



EUROPEAN SCHOOL OF MOLECULAR MEDICINE

NAPLES SITE – *Scientific Coordinator Prof. Francesco Salvatore*

UNIVERSITA' DEGLI STUDI DI NAPOLI "FEDERICOII"

Ph.D. in Molecular Medicine

Molecular Oncology Curriculum

XX Ciclo

*Identification of novel regulatory elements in
sequenced genomes by clustering and other data
mining methods*

Supervisor:

Prof. Francesco Salvatore

Ph.D. student:

Dr. Luca Cozzuto

Internal Supervisor:

Prof. Giovanni Paoletta

External Supervisor:

Prof. Toby Gibson

Index

INDEX	1
TABLE AND FIGURE INDEX	3
LIST OF ABBREVIATIONS USED	4
ABSTRACT	5
INTRODUCTION	6
GENOME ANNOTATION	6
COMPUTATIONAL METHODS IN FUNCTIONAL RNA DETECTION	7
SECONDARY STRUCTURE ANALYSIS	9
<i>Inverted repeat search</i>	11
<i>Maximization of pairing</i>	12
<i>The Minimum Folding Energy: Zuker algorithm</i>	14
<i>MFE evaluation and RNA structure</i>	17
<i>RNA families in structure prediction</i>	18
PATTERN SEARCH	20
SYSTEMATIC RNA SEARCH IN GENOMES	22
REPEATED STEM-LOOPS IN BACTERIA	24
LARGE-SCALE SEQUENCING IN BACTERIAL GENOME ANALYSIS	26
<i>The Scaffolding problem</i>	28
<i>Assembly and repeated sequences</i>	30
RESULTS AND DISCUSSION	32
FAMILIES OF STEM-LOOP STRUCTURES IN PROKARYOTIC GENOMES	32
<i>Finding repeats able to fold in a stem loop structure</i>	32
<i>SLS contained in repeats are able to fold in a stable way</i>	36
<i>Finding relations between clusters</i>	38
<i>Expanding detected repeated families by using Hidden Markov Model</i>	41
<i>Secondary structure analyses</i>	47
<i>Genomic localization of detected families</i>	49
<i>Characterization of specific families</i>	50
<i>Discussion</i>	59
GENOME ASSEMBLY BY “SCAFFOLDER”	66
<i>Finding links by using contig similarity and coding information</i>	66
<i>Finding links based on initial (raw) reads</i>	67
<i>Displaying relations as a connected graph</i>	71
<i>Graph analysis</i>	73
<i>Resolutions of ambiguities</i>	73
<i>Scaffolder tool</i>	80
METHODS	92
SELECTION OF HIGHLY REPEATED SLS	92
<i>Analyzing stability of SLS predicted secondary structure</i>	92
<i>Regrouping clusters in larger families</i>	93
<i>Extension of families members by cycles of HMM searches</i>	93
<i>Secondary structure analyses</i>	94
SCAFFOLDER	96
<i>Finding links between contigs</i>	96
<i>Building the connected graph</i>	96
<i>Support in design of PCR experiments</i>	97
<i>Aligning initial reads to a reference sequence</i>	98
<i>Micro-heterogeneities analysis</i>	98
REFERENCES	99
PAPERS	99
WEBSITES	105
ACKNOWLEDGMENTS	107

Table and figure index

Table 1. Functional RNA classes.....	8
Table 2. Organisms sequenced by pyrosequencing	28
Table 3. Sequence-based clustering of SLSs	34
Table 4. Regrouping of SLS clusters	40
Table 5. Families of SLS containing repeated sequences.....	46
Table 6. Secondary structure prediction analysis of families	49
Table 7. Structural properties of the SLS families in relation to genomic location	50
Table 8. Linked contig ends.....	69
Table 9. Contig coverage related to link number	70
Table 10. Options available by using the Scaffolder command line	88
Figure 1. RNA secondary structure	10
Figure 2. RNA secondary structure of a pseudoknot.....	11
Figure 3. Circular plot of a RNA secondary structure	13
Figure 4. Circular plot of a pseudoknotted secondary structure.....	14
Figure 5. Secondary structure decomposition	16
Figure 6. Average identity of detected clusters	35
Figure 7. Consensus lengths of detected clusters	35
Figure 8. Randfold analysis	37
Figure 9. SLS pipeline flowchart.....	43
Figure 10. Elongation process	44
Figure 11. ERIC family (<i>E. Coli</i>)	52
Figure 12. Sta-1 family (<i>S. aureus</i>)	53
Figure 13. Pae-1 family (<i>P. aeruginosa</i>)	54
Figure 14. Efa-1 family (<i>E. fecalis</i>)	55
Figure 15. Pu-BIME family (<i>S. typhi</i>)	56
Figure 16. dRS3 family (<i>N. Meningitidis</i>)	57
Figure 17. Myt-10 family (<i>M. tuberculosis</i>).....	58
Figure 18. Myt-1 (<i>M. tuberculosis</i>) and Pae-4 (<i>P. aeruginosa</i>) families	58
Figure 19. Finding links by BLAST	69
Figure 20. Link weight distribution	70
Figure 21. Genomic assembly of a 5.5 Mb bacterium as a connected graph	72
Figure 22. Alignment of a high coverage contig with primary reads.....	74
Figure 23. Solving a repeated contig by micro-heterogeneity analysis.....	76
Figure 24. Design of PCR experiment.....	79
Figure 25. Solving ambiguities by using PCR experiments	79
Figure 26. Assembly of a restricted number of contigs as a subgraph	82
Figure 27. Displaying contigs in tabular way	83
Figure 28. Scaffold history	85
Figure 29. Progression of the assembly of a 5.5 Mb bacterium in time	86
Figure 30. Web interface for Scaffolder (1)	89
Figure 31. Web interface for Scaffolder (2)	90
Figure 32. Management of PCR experiment results.....	91

List of abbreviations used

bp, base pair

CDS, coding sequence

CRISPR, clustered regularly interspaced short palindromic repeats

DUS, DNA uptake sequence

HMM, Hidden Markov Model

IS, insertion sequence

MCL, Markov Clustering algorithm

MFE, minimum folding energy

MIRU, mycobacterial interspersed repeated unit

nt, nucleotide

SCR, SLS-containing region

SLS, stem-loop-structure

TIR, terminal inverted repeat

Abstract

In bacterial genomes a fraction of transcribed sequences do not code for proteins or structural RNAs, but have been shown to be involved in fundamental processes, such as regulation of gene expression, mRNA processing and stability or structural RNA maturation. In this thesis a systematic procedure to identify and classify families of repeated sequences sharing a common RNA secondary structure was applied to the study of 40 bacterial genomes. Sequences able to fold in a stable stem loop structure were clustered according to sequence similarity, and grouped within homogeneous families. The study led to the identification of 57 families of repeated sequences, sharing a common secondary structure and potentially coding for structured RNAs. All previously known such families have been detected by the used procedure, and are listed within the final set, together with 37 novel ones. Their location in relation to protein coding genes was evaluated, and a correlation was found between structure and positioning within intergenic regions.

A new software tool is also described, Scaffold, designed to help in high-throughput *de novo* genome sequencing by finding connections between contigs produced by random shotgun sequencing, and assisting the researcher in the whole process. The software, accessible both as a command line tool and as a web application, can guide all the final phases of genome assembly by storing the current assembly status, displaying networks of connected contigs and untangling multiply connected ones by a combination of computational and experimental procedures.

Introduction

Genome annotation

Sequencing the human genome and that of other organisms created the conditions for sequence studies at the genomic scale, by allowing systematic analysis of the structure and organization of genomic regions. Genome annotation is a major challenge in genome projects and consists of identifying the location of known functional elements, such as genes and regulatory regions, as well as recognizing the role of unknown sequences, by attaching to them biologically relevant information.

The basic level of annotation relies on looking for sequence similarities into databanks containing known protein or DNA sequences such as Swiss-Prot or GenBank. Programs based on heuristic algorithms such as BLAST [Altschul et al 1990] are preferentially used in this kind of analysis. Genes may also be found by using predictive techniques. Searching for open reading frames (ORFs) in prokaryotes and other organisms characterized by uninterrupted genes allows gene identification in most cases: comparison of predicted ORFs with already described proteins allows to identify common structural proteins and enzymes involved in specific metabolic pathways. Databases such as KEGG [Kanehisa et al 2000] are used to evaluate which pathways are involved in a particular species or strain.

Gene detection in eukaryotes requires more complex procedures. The Ensembl project, born in 1991 to provide a centralized resource for researchers involved in genome analyses, uses a pipeline which first identifies the corresponding full-length cDNA for a given protein sequence retrieved by protein databases, and then detects the complete structure of transcript by aligning the cDNA to the genome. Finally expressed sequence tag (EST) collections are used to better define untranslated mRNA boundaries. Previously unknown genes may be identified by gene prediction programs based on various methods,

ranging from complex probabilistic models [Majoros et al. 2004] to neural network based exon detecting tools [Xu Y et al. 1994]. Integrated approaches such as in Genscan [Burge et al. 1997] are probably the best currently available, associating good sensitivity with low levels of wrong identifications.

In addition to genes, other functionally relevant sequences, such as protein binding sites or sites for attachment to nuclear scaffold, may be detected by using automated methods. Other interesting aspects can be evaluated by comparing sequences of closely or distantly related genomes such as orthologue genes and conserved sequences outside the coding portions. With the continuously increasing number of available complete genomes, use of automatic annotation methods is essential to quickly perform large-scale annotations aimed to detect functional genomic elements.

Computational methods in functional RNA detection

The discovery of several classes of functional RNAs in eukaryotes and the evidence that the majority of genome is transcribed but does not code for proteins [Kampa et al. 2004] stimulated bioinformaticians to develop new strategies able to detect these sequences by scanning the non-coding portion of the genome. Functional RNAs are molecules that exert their biological function at the RNA level, rather than through an encoded protein. They are known to be involved in plenty of biological processes such as gene expression regulation, post-transcriptional processing and maintaining chromosomal structure. In some cases such as antisense RNA (aRNA), their activity is only depending on their primary sequence, but, more often, their activity is connected to their three-dimensional structure.

Various classes of functional RNAs are reported in table 1, together with their main biological functions. Functional RNAs can be detected by sequence similarity by using BLAST like tools, but this procedure is often not useful where structure rather than

sequence defines the classification and role of a RNA molecule. This happens when structure rather than primary sequence is preserved during evolution, due to its direct involvement in biological function.

Currently two approaches can be followed to scan genomes looking for functional RNAs:

- 1) Identify by structure analysis sequences potentially able to fold in a stable structure.
- 2) Identify sequence and or structure patterns that are typical for a family of RNAs.

Name	Abbreviation	Function	Distribution
Ribosomal RNA	rRNA	Part of translation machinery	All
Transfer RNA	tRNA	Part of translation machinery	All
Signal recognition particle RNA	SRP RNA	Involved in protein trafficking	All
Transfer-messenger RNA	tmRNA	Rescues stalled ribosomes	Prokaryotes
Small nuclear RNA	snRNA	RNA maturation, regulation of gene expression and telomere maintaining.	Eukaryotes and Archea
Small nucleolar RNA	snoRNA	RNA maturation	Eukaryotes and Archea
Ribonuclease P	RNAseP	tRNA maturation	All
Ribonuclease MRP	RNAse MRP	rRNA maturation, DNA replication	Eukaryotes
Telomerase RNA		Telomerase synthesis	Eukaryotes
Antisense RNA	aRNA	Gene expression regulation	All
Cis natural antisense transcript	NAT	Gene expression regulation	Eukaryotes
Clustered Regularly Interspaced Short Palindromic Repeats RNA	CRISPR RNA	Host defense from parasites	Prokaryotes and Archea
MicroRNA	miRNA	Gene expression regulation	Eukaryotes
Piwi-interacting RNA	piRNA	Silencing of mobile elements	Animalia
Riboswitch		Gene expression regulation	All
Small interfering RNA	siRNA	Host defense from parasites, gene expression regulation	Eukaryotes
Y RNA		RNA processing, DNA replication	All
Group II intron		RNA maturation	All

Table 1. Functional RNA classes

A list of the main functional RNA classes is shown together with name, common abbreviation, main functions and distribution among different organisms

Secondary structure analysis

Nucleic acid folding can be considered as a multi-step hierarchical process in which a three-dimensional structure can be guessed starting from the secondary structure, which in turn is obtained by two-dimensional folding of the primary structure according to the rules of base pairing [Tinoco et al 1999]. This is due to the fact that interactions involved in forming secondary structure, basically the hydrogen bonds involved in classical A-T G-C Watson Crick and in G-U base pairing, are generally stronger than the additional ones involved in stabilizing tertiary structure. In principle, once the secondary structure is known, it is possible to infer the final structure by using the secondary structure information as a scaffold onto which the 3D structure in space is modelled.

Four types of secondary structure domains exist: helices, bulges, loops and junctions. Helices are Watson-Crick duplexes; loops, bulges and junctions are all unpaired regions terminated and defined by one or more helices (see Figure 1). Loops can be divided in internal and hairpin according to whether they are flanked by two helices or one. A bulge is a special case of internal loop, with no free base on one side and at least one free base on the other side, and a junction is the stretch of sequence connecting two adjacent structures. In addition pseudoknots may be formed when a loop is involved in the formation of a stem through base pairing with sequences located outside the loop itself (see Figure 2). Analysis of secondary structure deals with the correct recognition of most or all these domains in nucleic sequence.

Starting from these considerations in the last 30 years scientists have tried to design methods that allow the prediction of the secondary structure starting from nucleic acid sequence.

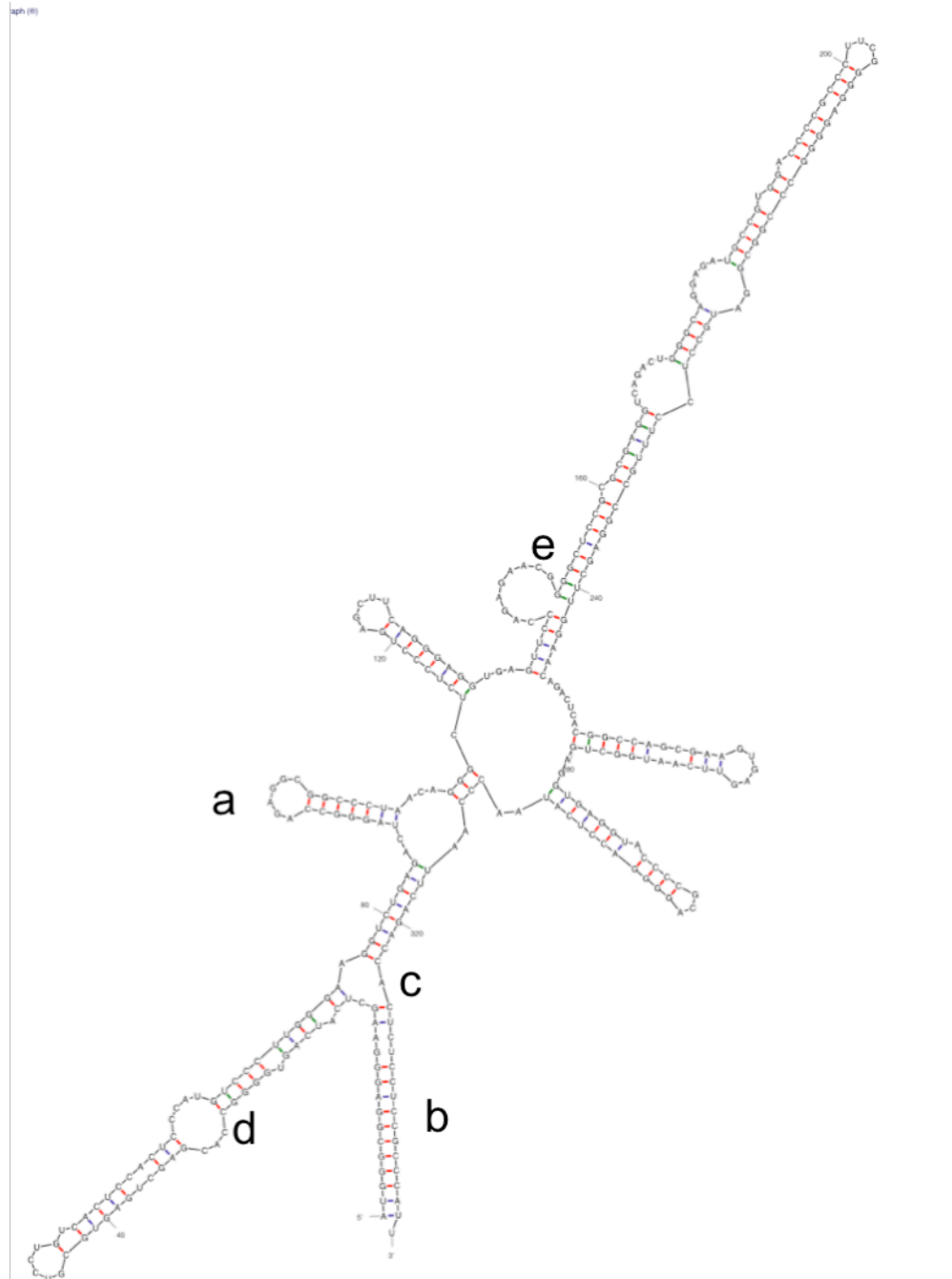


Figure 1. RNA secondary structure

The representation of the secondary structure of human RNA component (H1) of ribonuclease P is shown predicted by using the Mfold tool, that implements the Zuker algorithm. Base pair types are colored differently: GC in red, UA in blue, GU in green. Letters indicate different secondary structure domains: hairpin loop (a), stem or helix (b), multi branched loop (c), internal loop (d) and bulge loop (e).

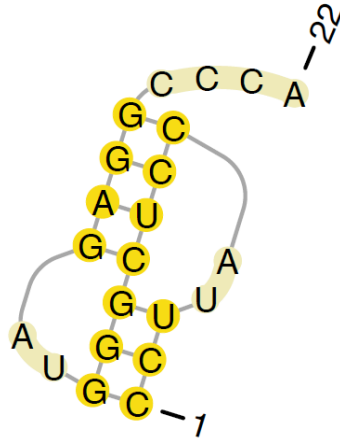


Figure 2. RNA secondary structure of a pseudoknot

The representation of the secondary structure of a pseudoknot domain is shown. Matching base pairs are highlighted in yellow within the structure. The secondary structure was created by using the Pseudoviewer software.

Inverted repeat search

A first step in secondary structure analysis is the identification of sequence regions able to fold as helices. Because the interactions involved in helix formation are the canonical Watson-Crick pairs (GC and AT or AU for RNA), they can be detected by looking for inverted repeats, i.e. aligning the sequence with its reverse complement by standard algorithm such as Needleman-Wunsch. Given two sequences **X** and **Y** it is possible to construct a scoring matrix $s(i, j)$ between all possible couple of bases of two sequences. It is possible to determine the best alignment with a “traceback” procedure that starting from the final part of alignment connects all matching bases in order to obtain the maximum score. This procedure is too simple to fully predict the secondary structure of an RNA but it is fast and can be useful for initial screening of long sequences to define putative boundaries of structured RNAs.

Maximization of pairing

A complication of the previously described procedure consists of looking for structures including maximum number of base pairs among all the possible ones. The Nussinov algorithm [Nussinov et al 1980] does this by giving the same weight to each base pair. The folding problem is considered as a variant of the maximum circular matching problem (MCMP) that has the scope to obtain for a circle the maximum number of chords without intersection. (See Figure 3).

Considering sequence **B**, composed by **B₁**, ... **B_n** nucleotides, that contains the subsequence **B_i**, **B_j** of length **p** with **j > i**, let **B_k** be a nucleotide between **i** and **j-1** positions. With a first recursion the algorithm tests the ability of **B_k** to pair with **B_j**, i.e. verifies if bases are AU or GC for each **k** position. With a nested recursion the algorithm calculates also the base pairs contained by subsequence delimited by **B_{k+1}** and **B_{j-1}** and **B_i** and **B_{k-1}**. After testing all **k** positions, the best value is saved in the matrix **M(i, j)**. If **B_j** cannot pair with any **k** then **M(i, j) = M(i, j-1)**. The maximum number of base pairs is obtained by incrementing **p** and repeating the search.

$$M(i, j) = \max \left\{ \begin{array}{l} M(i, k-1) + M(k+1, j-1) \\ M(i, j-1) \\ i \leq k < j = i + p \end{array} \right\}$$

The algorithm can consider, in addition to the standard Watson-Crick pairs (GC and AU), also the non standard GU pair often present in RNA structures. Once the scoring matrix is filled, it is possible to identify the secondary structure by a standard traceback procedure.

Although the search for stems is exhaustive, the solution found is often not unique, and the extreme simplicity of the scoring system may prevent reliable prediction of a correct secondary structure. An improvement of this method is the introduction of a scoring system based on the free energies associated with the formation of each base pair type, but even this thermodynamic model is not adequate to consistently predict correct secondary

structures. Moreover Nussinov algorithm cannot predict pseudoknots because base pairs occurring in these structures overlap with others, i.e. the representation of folding process as the MCMP is in contrast with the pseudoknot structure (see Figure 4). Since prediction of pseudoknots is computationally complex, most algorithms prefer to keep these structures out of their evaluated folding space. For this reason algorithms able to predict pseudoknots will not be described here.

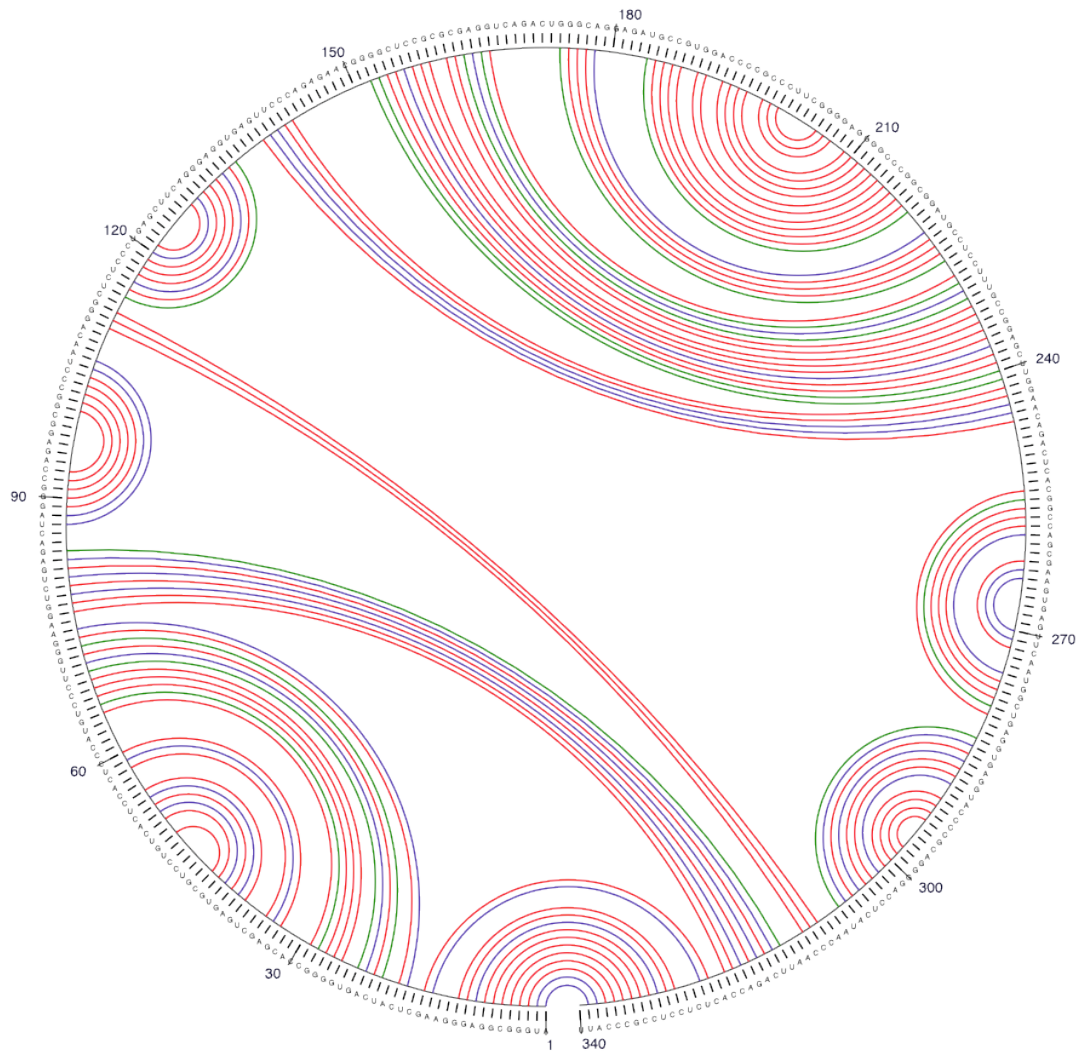


Figure 3. Circular plot of a RNA secondary structure

The representation of the secondary structure of the molecule shown in figure 2 is shown as a circular plot. The sequence is represented by a circle and base pairs as non-intersecting chords. Base pair types are colored differently: GC in red, UA in blue, GU in green. The prediction was made by using the Mfold tool.

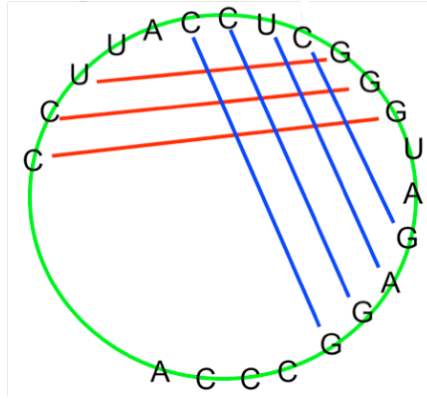
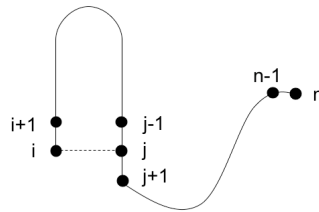


Figure 4. Circular plot of a pseudoknotted secondary structure

The representation of the secondary structure of the pseudoknot domain in figure 3 shown as a circular plot. The sequence is represented on a circle and base pairs as intersecting chords. Base pair types are colored differently to highlight base pair cross-links.

The Minimum Folding Energy: Zuker algorithm

Zuker and Stiegler developed in 1981 an algorithm, which is still probably the most frequently used today within the scientific community, to calculate the secondary structure starting from a single sequence.



Given a sequence of nucleotides $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ of length \mathbf{n} and energy parameters \mathbf{P}_{ij} , that describe the stability of the base-pair $(\mathbf{x}_i, \mathbf{x}_j)$, calculating the best structure means finding the structure with the lowest free energy.

By using the Nussinov algorithm this value, also called the minimum of folding free energy (MFE), is the lowest sum of base pair energies involved in the structure:

$$E = \min \left(\sum_{(i,j)} P_{ij} \right)$$

In the Zuker algorithm, this is improved by inserting additional corrections, such as one that avoids energetically unstable hairpin-loops shorter than three base pairs ($i > j + m$; with $m \geq 3$).

From a thermodynamic point of view the building blocks of secondary structures are all loops: stacked base pairs or helices, internal loops, hairpins and multi-branched loops are all interpreted as loops with a varying number of unpaired bases. (See Figure 5). The Zuker algorithm tries to determine which of the four elementary structures with the exterior pair (i, j), has the lowest free energy. A recursive approach is used to evaluate relations and produce a two dimensional matrix where all minimum free energies for each i and j is stored. Again backtracking is necessary to build the path that gives the MFE secondary structure.

To calculate energies, the algorithm uses the nearest-neighbor model, which assumes that the thermodynamic stability of a specific base pair depends on the neighboring bases. In this way both binding and stacking energies are evaluated at the same time.

The algorithm also takes into account the energy associated to each loop, delimited by bases $j+1$ and $j-1$, and to dangling ends delimited by $j+1$ and n :

$$E_{i,n} = \min \left\{ E_{i,n-1}; \min (E_{i+1,j-1} + E_{j+1,n} + P_{ij}) \right\}$$

Energy changes associated to various types of loop have been tabulated in relation to loop type and size and are used as energetic penalties. The energy values are derived from empirical calorimetric experiments and are minimized by a recursive procedure. $E_{1,n}$ is the minimum free energy for the full secondary structure involving the whole sequence X .

Tools implementing this algorithm have been shown to correctly recognize up to 65% of the base pairs of a structure [Gardner et al. 2004]. This number may be improved by introducing additional constraints derived from experimental information. For instance the flavin mononucleotide (FMN) is able to photocleave RNA specifically at U residues involved in G-U base-pairs: this information can directly be used to improve secondary

structure prediction. Various limits reduce the accuracy of this method: energy parameters calculated in laboratory are often slightly different from in vivo conditions and modified bases are ignored although they are known to have an important role in RNA secondary structure formation. Recently some chemical modifications involved in base pairs have been added to the table and used in evaluating the thermodynamic nearest neighbor model [Mathews et al. 2004].

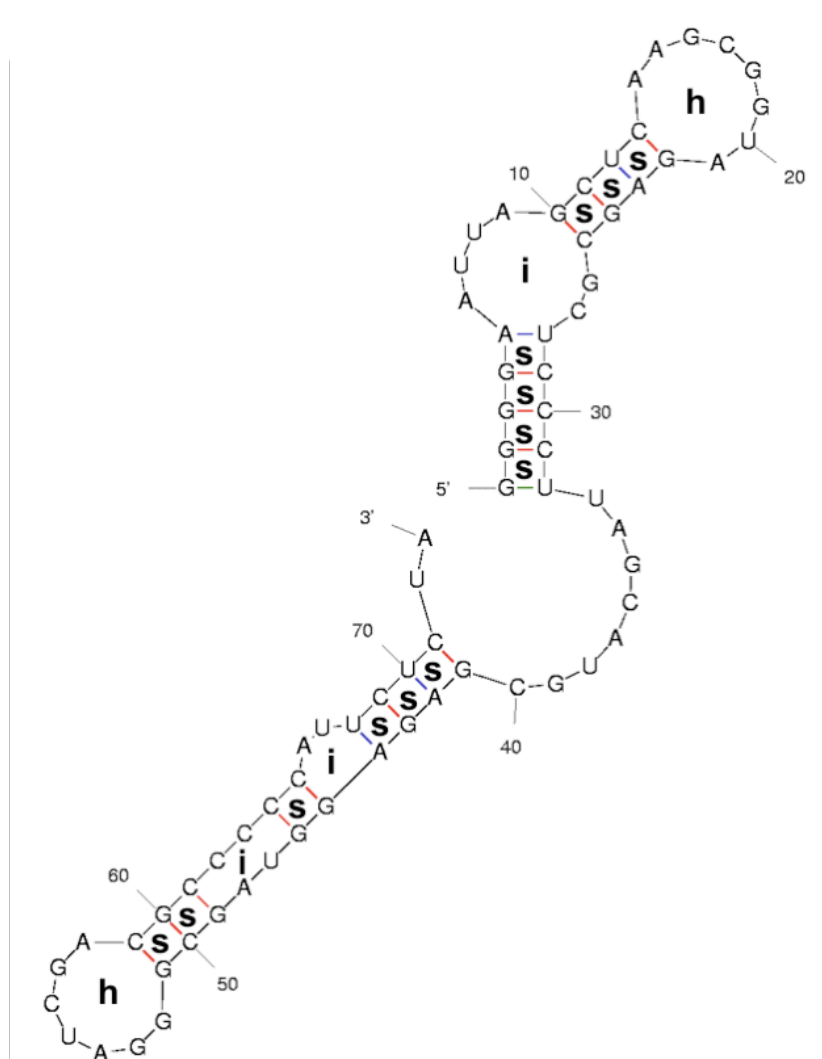


Figure 5. Secondary structure decomposition

The reported secondary structure is decomposed into loops delimited by two or more base pairs. Loops are indicated in this way: **h** for hairpin, **i** internal and **s** stacked.

MFE evaluation and RNA structure

In principle the Zuker algorithm should produce the optimal structure; most failures in prediction accuracy are more likely to be due to a scoring system's inaccuracy rather than an algorithm problem. The thermodynamic parameters are generally assumed to be accurate within a 5-10% range, but surprisingly an incredible number of alternative RNA structures lies in this interval. Moreover some RNAs have a bi-stable structure that cannot be predicted by looking for the MFE.

For these reasons, the correct structure might not be the one associated with the MFE, but rather one with a higher folding energy than the calculated MFE and therefore it cannot be revealed simply by energy minimization. Zuker proposed to also look at suboptimal structures [Zuker et al. 1989], and Wutchy et al. in 1998 developed a method to calculate the entire ensemble of suboptimal structures ranging between the MFE and an arbitrary upper limit. By using this approach, secondary structure prediction may include evaluation of several structures for a single sequence. Gardner et al. in 2004 tested tools implementing this algorithm on the ability to correctly recognize four kinds of known structured RNAs ordered by length: *S. cerevisiae* Phe-tRNA (73 bp), *E. coli* RNase P (377 bp), *E. coli* SSU rRNA (1542), and *E. coli* LSU rRNA (2904 bp). This work demonstrated that sensitivity and selectivity of these methods range from 22-63% and 20-60% respectively and that can rise to 22-69% and 20-67% by only investigating the first two suboptimal structures.

Even if algorithm is not completely accurate, the calculate MFE can be thought in principle as a good indicator of presence of structured functional RNAs in a genomic sequence. In practice, it can be used if some precautions are taken into account: obviously MFE should be normalized to sequence length, because the number of base pairs increases with the molecule size. Moreover structures derived from higher GC content sequences are likely to have lower MFEs than others, as a higher GC percent inevitably results into a larger

number of more stable GC pairs [Freyhult et al 2005]. A way to give better statistical significance to a MFE is to compare it with MFEs derived from analyzing a set of random sequences with the same length and nucleotide composition. Workman and Krogh in 1999 used the z-score, defined as

$$z = \frac{(E - \mu)}{\sigma}$$

where E is the MFE, μ is the average and sigma the standard deviation of the distribution of MFE values for a pool of random sequences. They found that in most cases mRNAs have a MFE undistinguishable from those obtained from randomized sequences, while a striking difference was found in the case of the highly structured ribosomal RNAs (rRNAs). Transfer RNAs (tRNAs), although structured, show MFE values similar to those obtained for random sequences, and cannot be easily identified by using the z-score indicator.

In 2004 Bonnet et al. analyzed the z-score of a recently discovered class of little functional RNAs: the micro-RNAs. They used a variant of the z-score procedure that makes no assumptions upon the nature of the MFE distribution, and demonstrated that more than 70% of known micro-RNAs show low z-scores. They used a Monte Carlo randomization test to calculate the probability (p) for a given sequence to fold better than random ones obtained by reshuffling of the sequence itself:

$$p = \frac{R}{N + 1}$$

where R is the number of random sequences with a MFE less or equal than the original and N represents the total number of random sequences. Currently z-score is believed to be a good indicator for structured RNAs, in particular long stems, although not very sensitive.

RNA families in structure prediction


The limited accuracy of RNA structures predicted on the basis of single sequence folding

suggested the need for further biological information to improve the predictions, such as that derived from comparative analyses. Three different approaches to predict secondary RNA structure by using comparative RNA sequence analysis have been developed:

- use pre-aligned nucleotide sequences to infer a common secondary structure
- try to simultaneously align and infer a consensus secondary structure
- directly align RNA structures derived from folding prediction.

The first approach is used in the algorithm proposed by Hofacker et al. in 2002.

	i	j
α	A	N	N	N	T
β	A	N	N	N	T
γ	G	N	N	N	C
...					
N	C	N	N	N	G



Considering nucleotides $\mathbf{a_i}$ and $\mathbf{a_j}$ at each row ($\alpha, \beta, \gamma, \dots \mathbf{N}$) of a sequence alignment \mathbf{A} , new energy parameters $\mathbf{P^A_{ij}}$ are calculated by combining the average pairing energy of $\mathbf{a_i}$ and $\mathbf{a_j}$ with the covariance score $\mathbf{C_{ij}}$ derived from the analysis of compensatory mutation.

$$P_{ij}^A = \frac{1}{N} \sum_{\alpha} \in (a_i^{\alpha}, a_j^{\alpha}) - C_{ij}$$

The algorithm performs much better than the single sequence folding method previously described, achieving sensitivity higher than 70% [Gardner et al. 2004]. Of course the intrinsic limit of this approach is related to the quality of the alignment. When identity is lower than 70%, incorrect sequence alignments can destroy the co-variation signal.

The second method is based on algorithm described by Sankoff in 1985, which aims to obtain a common base-pair list which maximizes the sum of base-pair weights. Because the original algorithm is computationally very expensive, variants containing particular restrictions have been implemented [Gorodkin et al. 1997, Mathews et al. 2002]. However up to now they have been only able to detect a fraction of present pairs [Gardner et al.

2004].

The third approach to predict secondary structure is based on aligning the RNA structures in order to detect the best common one, independently or with limited dependence on the sequences. This can be done, for example, by implementing the tree alignment model [Höchsmann et al. 2003]. Obviously the results of this approach are strictly related to the quality of the prediction of the single structures. This approach is used at its best when individual predictions are made by the first method, starting from a family of closely related sequences, and then compared with other molecules belonging to families that have different sequences but similar structure.

Pattern search

The main limit of using structure prediction to search for functional RNAs is related to the stability of the searched secondary structure. The long structured stems of H/ACA snoRNAs and miRNAs are often not difficult to spot, but smaller unstable stems like those contained in C/D snoRNAs are easily missed [Washietl et al. 2005]. Moreover, when functional RNAs are not conserved or no genome related to the analyzed one has been sequenced, these methods cannot provide the best results because of the absence of covariance information. These conditions are often found in bacteria, where conservation is limited or absent in phylogenetically distant species.

In order to overcome these problems, specific tools have been designed, aimed to detect a particular functional RNA class such as tRNA, C/D and H/ACA snoRNAs, tmRNA, miRNA [Lowe et al. 1997, Lowe et al. 1999, Laslett et al. 2002, Schattner et al. 2004, Lim et al. 2003]; these tools depend on specific RNA features that are often combinations of sequence and structure motifs. More general strategies are based on pattern search and on covariance models. Pattern search is implemented in very customizable tools, such as RNAMotif [Macke et al 2001], that allows to selectively detect functional RNAs sharing

structural and sequence characteristics typical of a specific class of RNAs. This approach may be very effective in the right context, but of course cannot be used to discover new classes of functional RNAs.

The covariance model (CM), described for the first time by Eddy in 1994, [Eddy et al. 1994] is a probabilistic model for analysis of RNA secondary structures, analogous to sequence search by profile hidden Markov model. A CM is built starting from a sequence alignment and a consensus structure and can be used to scan entire genomes. The extreme slowness of tools implementing this algorithm requires the use of powerful computational resources to search a single structure in a single eukaryotic genome [Klein et al. 2003].

CM can be also used to detect new undescribed structures as in the algorithm proposed by Yao et al. [2005], that it is used to find structured motifs in unaligned but evolutionary related sequences. The algorithm first identifies a group of subsequences with the lowest MFEs and then it uses a tree-editing algorithm to iteratively align them in order to find the consensus structure. To improve the efficiency, the alignments are limited to sequences compatible with locally conserved regions found by BLAST search. The best 10 alignments are used as seeds to the expectation maximization algorithm that predicts the RNA secondary structure by using a CM.

Other strategies based on analyses of substitution patterns and RNA structure modelling have been implemented. Pedersen et al. developed a procedure based on two competing phylogenetic–stochastic context-free grammar (phylo-SCFG) models of RNA sequence evolution: a structural model and a nonstructural model [Pedersen et al. in 2006]. Structure is only predicted when a segment of the alignment is better described by the structural model than the nonstructural model. The two models describe alignments with identical properties, except that the nonstructural model assumes a higher substitution rate and does not include correlated base-pair changes, as found in RNA helices. To each structure prediction a score is assigned based on the relative likelihood of the alignment under the

combined structural/nonstructural model and a purely nonstructural model. This approach has been demonstrated to work for tRNAs and microRNA detection but not on snoRNAs [Pedersen et al. in 2006].

Systematic RNA search in genomes

Different attempts to perform systematic screenings, looking for functional RNAs, have been done in recent years. In 2001 Rivas and Eddy, compared intergenic sequences of two related bacteria *E. coli* and *S. typhi*, in order to detect putative structured RNAs. They used an algorithm based on the covariance model, that is able to compare only two aligned sequences. The strategy allowed detecting 275 candidate structural RNA loci that have been checked in part for their ability to be transcribed as small non-coding RNAs. 11 out of 49 loci predicted to be structured have been shown to be transcribed. Interestingly some of these positive sequences belong to a class of already described DNA repeats, sharing a conserved palindrome called BIMEs (Bacterial Interspersed Mosaic Elements), known to be involved in a variety of biological processes thanks to their RNA structure [Bachelier et al. 1999]. In 2005 Berezikov et al., in order to find conserved micro-RNAs, focused their attention on sequences conserved across different eukaryotic genomes. They first selected conserved sequences, predicted to fold in a stem-loop structure by using tools implementing the Zuker algorithm and then evaluated their MFEs by calculating the z-score with the procedure proposed by Bonnet [Bonnet et al. 2004]. In this way, they detected 379 putative micro-RNAs that are conserved across human, mouse and rat genomes. 119 of them resulted to be already described and correctly recognized.

In the same year, Washietl et al. used MFE, calculated by the Hofacker algorithm, to detect the presence of structured functional RNA in aligned sequences. The method is based on comparison of pre-aligned sequences and contemplates the combination of two scores: the structure conservation index (SCI) and the average of Z-score of single sequences that

indicates the thermodynamic stability. SCI is defined as the ratio between the consensus MFE (E_A) and the average of MFE of each alignment sequence (E):

$$SCI = \frac{E_A}{\frac{1}{N} \sum E}$$

SCI values are around 1 for sequences sharing both primary and secondary structure similarity, but are increased beyond 1 for sequences where secondary structure is better conserved than sequence, due to compensatory changes. Sequences not sharing a secondary structure would produce very low scores, even down to 0. This method shows high sensitivity and specificity when finding several classes of functional RNAs characterized by conserved sequence and structure, such as tRNA, miRNA and some snoRNAs.

Washietl et al. performed a genomic screening by using the above described procedure on sequences derived from whole-genome alignment of eukaryotic species such as human, chimp, mouse, dog, chicken, zebrafish and fugu and predicted more than 30,000 functional RNAs, about 1,000 of them conserved across all vertebrates (Washietl et al. 2005). A second screening was conducted on the regions of the human genome analyzed by the ENCODE consortium, that also contain not conserved sequences, by using both the above described procedures by Washietl and by Pederson. The screening identified thousands of putative conserved functional RNAs [Washietl et al. 2007], but the structures identified by the two approaches show little overlap (< 8%). This probably reflects the fact that the Washietl approach is sensitive to alignments with moderate and high GC content and relatively low sequence similarity, while the other is sensitive for low GC content and high sequence similarity even if this generates many false positive results [Washietl et al 2007]. The same authors estimated that high false positive ratios, respectively 50% and 70% for Washietl and Pederson methods, are obtained by taking into account dinucleotide frequencies, when analyzing shuffled alignments. A small fraction of the predicted RNAs

was validated by RT-PCR in six tissues: RNA expression was confirmed in about 25% of cases.

The procedure described by Yao [2005] was also used in two genomic screenings. The first analyzed potential 5' UTR of conserved bacterial genes [Weinberg et al. 2007] and detected 22 putative structured motifs, some of them recognized as new riboswitch classes. The second was carried out on the same genomic regions analyzed by Washietl et al. in 2007 [Torarinsson et al. 2007] and predicted more than 6,500 structured loci, that only partially overlap with the results obtained in the previous screenings, thus extending the number of detected candidate functional RNAs by 32%. This increment is also due to the fact that alignments featuring many gaps or low sequence conservation and discarded by the previous methods are correctly detected by this procedure. Also for this method a relatively high false positive rate was estimated, about 50%.

Repeated stem-loops in bacteria

Although bacterial genomes are in general more compact than eukaryotic ones, with over 90% devoted to coding for protein genes, about 10% of DNA is still present as intergenic in prokaryotic genomes, and contains sequences coding for functional RNAs as tRNA and rRNA, but also less well defined types. Many functional RNAs are present in multiple copies in bacterial genomes, and studies on DNA repeats have often ended up by identifying families of transcribed sequences potentially coding for structured RNAs. Some of these repeats show a complex conserved secondary structure, that is clearly related to their activity, as in the case of self-splicing introns [Martínez-Abarca et al. 2000]. In others a conserved secondary structure has been observed, but is not clearly connected to a specific functionality, as in the case of a large class of repeated DNAs containing palindromes found in enterobacteria [Bachelier et al. 1999].

This class of repeats is comprised of sequences shorter than 200 bp, located in intergenic

regions and potentially transcribed but not generally coding for proteins. Their degree of repetition ranges between 10 and 500 copies in different bacterial species. Members of this class include *V. cholerae* VCR [Rowe-Magnus et al. 2003] and *E. coli* and *S. typhimurium* BIMEs [Engelhorn et al. 1995, Espéli et al 1997, Gilson et al. 1991]. Other palindromic, stem-loop containing repeats from the same class are RSAs and ERICs (or IRUs), simple repeats that have been found in *E. coli*, *S. typhi*, *K. pneumoniae* and *Y. pestis*. Also these are located in intergenic regions, in either orientation with respect to replication and transcription. Compensatory mutations observed in these families suggest a conserved secondary structure, possibly involved in functional roles such as translation interference or mRNA protection from digestion. Other repeats like BOCEs have been found in *E. coli* and *K. pneumoniae*. Overall their functional roles are not clearly defined, but in some cases, following experimentally studies, putative functions have been proposed for specific repeats. Some members of the BIME family were demonstrated to be involved in biological processes as transcription termination, gene expression regulation and protein interaction, possibly because of their stem-loop structure [Bachelhier et al 1999].

A systematic analysis of sequences able to fold as a stem-loop structure was attempted in 40 wholly sequenced bacterial genomes [Petrillo et al. 2006]. In order to reduce the number of possible structures, work was focused on those containing stems at least 12 base pair long. Comparison of SLSs contained within genomes with those obtained from random genomes demonstrated that natural SLSs are always more than those expected by chance. Moreover specific SLS subsets are found to be selectively enriched in natural genomes. SLSs with low MFEs (< -15 Kcal) and those with the smallest loops appear to be more frequent than expected and are hypothesized to be involved in formation of secondary structures, as those found in self-splicing introns [Martínez-Abarca et al. 2000], riboswitches [Nudler et al. 2004], and in the previously mentioned class of transcribed intergenic repeats including *E.coli* BIME, *Yersinia* ERIC and *Neisseria* NEMIS. In these

cases the stem is often essential to the attainment of the correct secondary structure and may be directly recognized by ribonucleases [Coburn et al. 1999, Gilson et al. 1991, De Gregorio et al 2005].

Large-scale sequencing in bacterial genome analysis

The search for functional sequences within complete genomes, is strongly dependent on the availability of large masses of genomic sequences. As far as the prokaryotic world is concerned, the complete DNA sequence of over 500 bacterial strains is known today and more are becoming available every month, from over 3000 bacterial genome sequencing projects. An important boost to these numbers is expected to come from the recent development of new DNA sequencing technologies such as pyrosequencing and hybridization sequencing, respectively used by commercially available high-throughput genome analyzers such as Roche 454 GS and Illumina.

Although these high-throughput techniques look very promising, most currently available sequences have been produced by using the standard Sanger method, and today only about 70 bacterial genomes have been sequenced by using the high throughput approach based on pyrosequencing (see table 2). By closely looking at the table, it appears that 4 of them are re-sequencing of already sequenced genomes, 52 are de novo sequencing of strains that can take advantage of information derived by related already sequenced genomes and 18 are “real” de-novo sequencing. Even among these last genomes, 7 were sequenced via a combination of high throughput and Sanger sequencing, and 11 by exclusively using the pyrosequencing approach. Of these, only 4 were completely sequenced and assembled, yielding a single genomic sequence, ranging in size between the 250 kilobases of *Candidatus Sulcia muelleri* and the 3.9 megabases of *Acinetobacter baumannii*.

Organism	N. genome	Size	Type	Technology	Complete
<i>Escherichia coli K12</i>	1	4,6	resequencing	454	yes
<i>Chlamydia trachomatis</i>	1	1	resequencing	454	yes
<i>Saccharopolyspora erythraea</i>	1	8,2	resequencing	454	yes
<i>Mycobacterium tuberculosis</i>	1	4,4	resequencing	454	yes
<i>Myxococcus xanthus</i>	3	9,14	strain	454+Sanger	yes
<i>Staphylococcus aureus</i>	2	2,8	strain	454+Sanger	yes
<i>Campylobacter jejuni</i>	1	1,6	strain	454+Sanger	yes
<i>Salmonella Typhi</i>	19	5	strain	454+Solexa	no
<i>Vibrio cholerae</i>	1	4,1	strain	454	no
<i>Campylobacter jejuni</i>	1	1,6	strain	454	no
<i>Escherichia coli O157:H7</i>	2	6,2	strain	454	no
<i>Helicobacter pylori</i>	2	1,6	strain	454	no
<i>Sinorhizobium meliloti</i>	1	3,6	strain	454	no
<i>Haemophilus influenzae</i>	9	1,8	strain	454	only 2
<i>Campylobacter jejuni</i>	1	1,6	strain	454	yes
<i>Streptococcus pneumoniae</i>	8	2,1	strain	454	yes
<i>Chlamydia trachomatis</i>	1	1	strain	454	yes
<i>Brucella abortus</i>	1	2,1+1,1	strain	454	yes
<i>Mycobacterium avium paratuberculosis</i>	1	?	<i>de novo</i>	454+Sanger	no
<i>Bacillus coahuilensis</i>	1	3,4	<i>de novo</i>	454+Sanger	no
<i>Bacillus pumilus</i>	1	3,7	<i>de novo</i>	454+Sanger	yes
<i>Acaryochloris marina</i>	1	6,5	<i>de novo</i>	454+Sanger	yes
<i>Corynebacterium urealyticum</i>	1	2,4	<i>de novo</i>	454+Sanger	yes
<i>Uncultured Termite group 1 bacterium</i>	1	1,1	<i>de novo</i>	454+Sanger	yes
<i>Acinetobacter baumannii ACICU</i>	1	3,9	<i>de novo</i>	454+Sanger	yes
<i>Beggiatoa</i>	2	7	<i>de novo</i>	454	no
<i>Vibrio furnissii</i>	1	?	<i>de novo</i>	454	no
<i>Acidimethylosilex fumarolicum</i>	1	?	<i>de novo</i>	454	no
<i>Corynebacterium kroppenstedtii</i>	1	2,4	<i>de novo</i>	454	no

Organism	N. genome	Size	Type	Technology	Complete
<i>Francisella tularensis</i>	1	2	<i>de novo</i>	454	no
<i>Campylobacter jejuni subsp. jejuni</i>	1	1,8	<i>de novo</i>	454	no
<i>Acinetobacter baumannii</i>	1	3,9	<i>de novo</i>	454	yes
<i>Candidatus Sulcia muelleri</i>	1	0,25	<i>de novo</i>	454	yes
<i>Pseudotriconympha grassii</i>	1	1,1	<i>de novo</i>	454	yes
<i>Oligotropha carboxidovorans</i>	1	3,7	<i>de novo</i>	454	yes

Table 2. Organisms sequenced by pyrosequencing

Organisms sequenced by 454 sequencers based on pyrosequencing technology are shown together with number and size of genome strains, type of sequencing, technology used and project state.

The Scaffolding problem

Large-scale whole genome shotgun sequencing was successfully applied for the first time in 1995 to determine the complete genome sequence of *Haemophilus influenzae* [Fleischmann et al. 1995], a 1.8 Mb bacterium, and subsequently used for many other bacterial strains, as well as for eukaryotic genomes. Shotgun sequencing consists of randomly breaking the genome into a large number of overlapping small fragments and sequencing them; final assembly of the fragments produces the complete sequence, typically with the help of an assembler tool. In the 1990s Phrap was probably the most frequently used assembler tool. It is based on a three-step procedure where after finding the best alignment for each matching pair of reads having more than one significant alignment in a given region, layouts of contiguous sequences are built, and finally contig sequences are generated as a consensus of the highest quality parts of the reads by using consistent pair-wise matches. This approach proved to be highly successful and was largely used for assembling the human genome. Unfortunately the algorithm expects relatively large primary reads (500-1000 bases) and is not adequate for the short reads generated by high

throughput sequencing machines, which are typically shorter (40-200 bp). This novel kind of sequencing has been defined “short read sequencing (SRS)” and required the development of a new class of programs, able to combine millions of very short reads. Two commonly used assembler tools are Newbler [Margulies et al. 2006], developed by 454 Life Sciences, a Roche owned company, and Euler-SR [Chaisson et al. 2008].

Newbler consists of a series of modules that act in subsequent steps, in a fashion similar to Phrap. First, the “Overlapper” module finds and creates all pairwise overlaps between reads. In the second step the “Unitigger” module constructs larger sequences containing overlapping consistent reads that are uncontested by reads external to the sequence. For this reason the obtained sequences are called “unitigs”. In the third step, the “Multialigner” module takes all the reads that make up the unitigs and aligns all the read signals generating a consensus sequence and quality scores for each base within each assembled “contig”.

Euler-SR is based on a different strategy from the “overlap-layout-consensus” approach implemented by Phrap and Newbler. It transforms the assembly problem into an Eulerian path problem by dividing all reads into overlapping k-tuples that become the vertices of a de Bruijn graph [Chaisson et al. 2008]. K-tuples are connected by links if they share a common segment of at least k-1 bases. The search for a unique ‘Eulerian’ path allows to create the final sequence. In most cases several independent paths can be identified allowing the assembly of different contigs. In many contigs, the presence of more than one link prevents the extension of the contig, given that more than one path do exist, passing through the contig, and creating a tangle in the global graph that is diagnostic of the presence of repeated sequences. Information on the reads can be used to untangle most of these cases but of course repeats larger than the read length cannot be solved. A comparative test of the two methods was carried out by assembling *Streptococcus pneumoniae* genome, sequenced by using reads shorter than 120 bases [Chaisson et al.

2008]. The genome is known to contain 167 exact repeats longer than 120 bases and is not resolvable by any assembler, as fragment assembly should theoretically generate 504 contigs, 136 of which larger than 500 bases. The ideal assembler should recognize all these 136 large contigs. This analysis revealed that Newbler manages to detect 255 contigs longer than 500 bases, collectively covering about 2000 kb while Euler-SR almost correctly identifies 127 long contigs, together covering 2001 kb.

Only in unusual circumstances these programs are expected to produce a single final assembled sequence; more often they generate a collection of contigs, whose location relative to each other or within the genome is not defined. For this reason sequencing is often complemented by a further procedure called “scaffolding”, necessary to order and orientate contigs by using other experimental data, such as long-range connectivity information.

Assembly and repeated sequences

The main cause that prevents the final assembly is the presence in genomes of repeated sequences, larger than the average read length. Because of this, the assembler software is often unable to separate and univocally assign those sequences to different contigs. Moreover some sequencing procedures require masking the repeated sequences and cause a sizable fraction of the genome not to be available within the final complete sequence. To overcome the problem Sundquist et al. in 2007 proposed a hierarchical sequencing strategy, called SHRAP (Short Read Assembly Protocol), based on sequencing multiple copies of the genome sheared and inserted in large fragment libraries, for example BAC clones, by SRS. Reads coming out from sequencing experiments are used to infer positioning of the clones along the genome according to clone maps generated in a pre-assembly step. The assembler tool is then used to sequence individual ordered clones. Tests using simulated data show that the SHRAP strategy is able to assemble large

genomes such as human or *D. Melanogaster*, but no trial with real experimental data have been performed yet. Some assembler tools include a scaffolding step that consists in using mate pairs data. In 2004 Pop et al. developed a general-purpose tool able to guide the scaffolding process called Bambus [Pop et al. 2004]. This tool is currently used in all sequencing project at TIGR and can manage several kinds of linking information such as mate information, homology data, physical maps and gene syntenies, presented as a connected graph.

Results and discussion

Families of stem-loop structures in prokaryotic genomes

Finding repeats able to fold in a stem loop structure

Sequences analyzed in this study derived from a previous work [Petrillo et al 2006], in which the analysis of complete genomes of 40 bacterial genomes, mostly of medical interest, predicted more than 5 million sequences as able to fold in a RNA stem-loop structure (SLS). SLS was defined as a structure with a stem of at least 12 bp, loop size ranging from 5 to 100 nucleotides and in which GU pairing is admitted. Sequences predicted to fold with a MFE lower than -5 Kcal/mol were selected for this study, with the exception of those falling within either mature RNA species (tRNAs, rRNAs) or known Inserted Sequences (IS), in order to avoid known structured repeated sequences. In this way the SLS population was reduced to slightly over 2 millions sequences.

Clustering

The SLS population was screened for the presence of repeats by clustering them according to sequence similarity. Sequence comparison was performed by running an all-against-all BLAST within the SLSs of each genome, and the resulting matches were used for the compilation of distance matrices in which the E-value is used as a measure of distance. BLAST was run without searching for the complementary strand, as in this step the goal was to identify similarity between the putative RNAs. In order to limit the selection to highly similar sequences, this clustering step was performed by using stringent parameters; in order to avoid clustering of SLS containing sequences on the basis of contiguity rather than content similarity, connections caused by overlapping sequences were eliminated (see

Methods). Clustering was done by feeding the resulting matrix to MCL [Enright et al. 2002], a tool implementing the Markov Clustering algorithm for unsupervised clustering, based on simulation of stochastic flow in graphs. Within MCL, the distance matrix is interpreted as a connected graph, where sequences are nodes and similarities are edges. As a consequence, groups of nodes characterized by the presence of many connecting edges represent clusters of similar sequences. Nodes belonging to a cluster are connected by paths that are typically more numerous and of better quality than those between nodes lying in different clusters. MCL uses random walking as a means to achieve cluster separation, since walking on paths within a cluster is far more likely than walking on paths connecting different clusters. Two operations, expansion and inflation, are iteratively performed on the matrix in order to progressively increase cluster separation.

By applying this technique, 523 clusters were identified, composed of at least 7 non overlapping genomic elements, as reported in Table 3. Although links between overlapping SLSs were removed, a small number of members of the same cluster were still found to map onto the same genomic sequence and were joined into larger SCRs, for SLS containing regions. Together, the 523 identified clusters, contain 12,254 non-overlapping SCRs corresponding to a total of 28,904 SLS elements, corresponding to about 1.3% of the originally selected SLS population. Individual clusters contains between 8 and over 4,000 SCRs.

Of the 40 analyzed genomes, 29 contain at least one and up to 75 clusters. No clusters were identified for the remaining 11 genomes: *L. innocua*, *L. monocytogenes*, *S. pyogenes*, *C. pneumoniae*, *C. trachomatis*, *U. urealyticum*, *R. prowazekii*, *T. pallidum*, *Buchnera*, *C. jejuni* and *H. pylori*. The quality of the described clustering procedure was evaluated by aligning SCR members of each cluster by the PCMA multiple alignment tool [Pei et al. 2003], and analyzing the resulting alignments by using ALISTAT [Bateman et al. 1999].

Division	Species	SLSs	Clusters	Clustered SLSs	Clustered SCRs
low-GC Firmicutes	<i>Bacillus anthracis</i>	65,220	4	105	38
	<i>Bacillus halodurans</i>	55,624	6	182	93
	<i>Bacillus subtilis</i>	56,622	2	32	16
	<i>Clostridium perfringens</i>	35,027	6	149	81
	<i>Clostridium tetani</i>	29,883	14	178	123
	<i>Enterococcus faecalis</i>	40,991	7	317	142
	<i>Lactobacillus johnsonii</i>	25,668	3	173	26
	<i>Staphylococcus aureus</i>	32,372	11	275	144
	<i>Streptococcus pneumoniae</i>	25,095	28	825	386
Mollicutes	<i>Mycoplasma genitalium</i>	8,953	1	21	8
	<i>Mycoplasma pneumoniae</i>	13,926	20	372	165
high-GC Firmicutes	<i>Corynebacterium diphtheriae</i>	54,254	9	282	120
	<i>Mycobacterium leprae</i>	83,094	29	1,721	537
	<i>Mycobacterium tuberculosis</i>	170,502	59	2,182	636
α -Proteobacteria	<i>Brucella melitensis</i>	69,899	11	399	219
	<i>Rickettsia conorii</i>	14,933	19	797	383
β -Proteobacteria	<i>Bordetella bronchiseptica</i>	214,459	26	2,009	470
	<i>Bordetella parapertussis</i>	188,237	30	1,513	518
	<i>Bordetella pertussis</i>	158,592	52	7,212	4,602
	<i>Neisseria meningitidis</i>	56,605	44	3,595	991
γ -Proteobacteria	<i>Escherichia coli</i>	86,339	12	1,152	431
	<i>Haemophilus influenzae</i>	25,055	3	39	25
	<i>Pasteurella multocida</i>	31,209	1	24	8
	<i>Pseudomonas aeruginosa</i>	206,492	9	526	129
	<i>Pseudomonas putida</i>	175,088	75	3,640	1,352
	<i>Salmonella typhi</i>	90,027	8	177	116
	<i>Salmonella typhimurium</i>	91,844	7	157	94
	<i>Vibrio cholerae</i>	45,824	7	250	122
	<i>Yersinia pestis</i>	78,372	20	600	279
TOTAL		2,230,206	523	28,904	12,254

Table 3. Sequence-based clustering of SLSs

BLAST-MCL based clustering of SLSs from bacterial genomes described in Petrillo et al 2007. For each species, the number of elements within the starting populations, the number of clusters and the number of clustered SLSs are reported. The number of SLS containing regions (SCRs), obtained by fusing overlapping clustered SLSs, is also reported. Only species featuring at least one cluster, with a minimum of 7 SCRs, are listed.

The analysis revealed that over than 80% of the clusters show an average identity higher than 60% and that the established consensus was larger than 90 bp for the about half of them, while the others produced consensus sequences between 27 and 90 bp (see Figures 6 and 7).

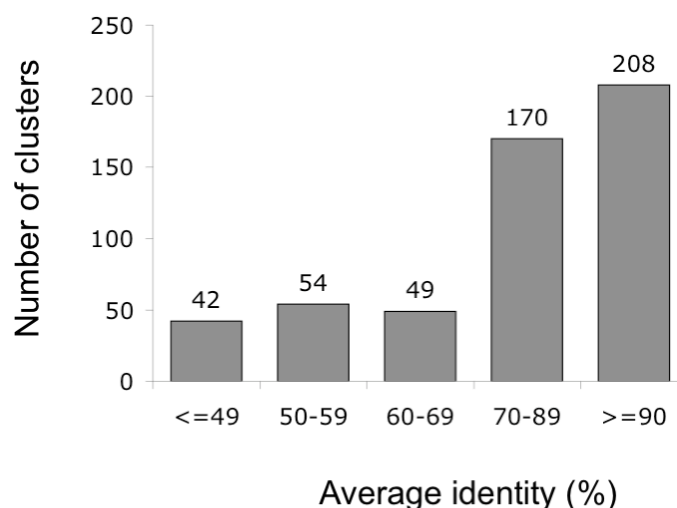


Figure 6. Average identity of detected clusters

In the graph bars represent the number of clusters falling within the reported average identity range. Members of each clusters were aligned by PCMA and alignment was evaluated by ALISTAT tool.

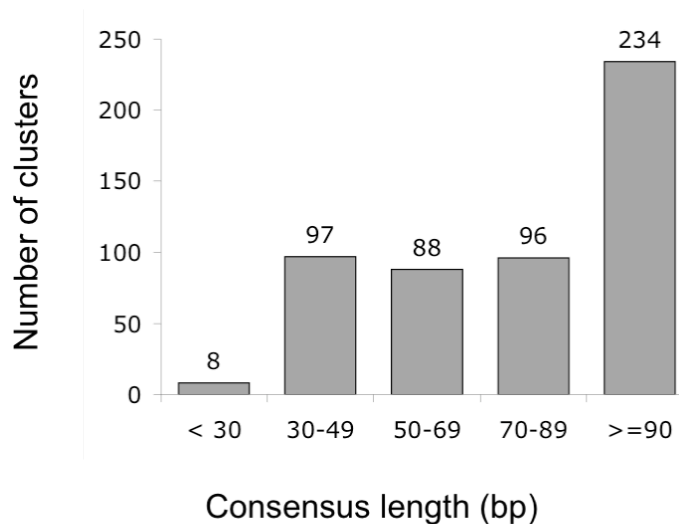


Figure 7. Consensus lengths of detected clusters

In the graph bars represent the number of clusters falling within the reported consensus length range. Alignment of members of each clusters was fed to ALISTAT tool to calculate consensus.

SLS contained in repeats are able to fold in a stable way

Clusters of similar SLSs were analyzed for their ability to fold into a reliable secondary structure, by using the procedure implemented by the RANDFOLD tool [Bonnet et al. 2004]. This procedure compares the predicted minimum folding energy (MFE) of a sequence with those of a large number of random shuffles of the same sequence. Results are expressed as a p-value, indicative of the predicted MFE being truly different from the others. Since predicted stability of RNA secondary structure is calculated on the basis of a nearest neighbour model, which also includes a base stacking component, sequences analyzed in this test were shuffled by preserving dinucleotide frequencies, as proposed by Workman and Krogh in 1999.

For each genome, RANDFOLD was run on three different sequence populations:

- SLSs clustered as described above;

- SLSs randomly picked from the initial population;

- Random genomic sequences of the same size as clustered ones.

The results obtained for each of these populations are reported in figure 8. Sequences belonging to each group are assigned to a specific “folding aptitude” class according to the p-values calculated by using RANDFOLD. Most SLSs obtained by the clustering procedure (panel A) show a non-random probability of folding lower than 0.01 (dark grey bars), and, very often, also lower than 0.001 (black bars), whereas only about 20% of the SLS from the original population reach these p-values (Figure 8, panel B). Only in four genomes, *M. leprae*, *L. johnsonii*, *M. genitalium* and *M. pneumoniae*, the two SLS populations do not show statistically different folding aptitudes. A very small fraction (less than 5%) of control sequences showed a non-random folding probability higher than 0.1% (light grey bars in Figure 8, panel C).

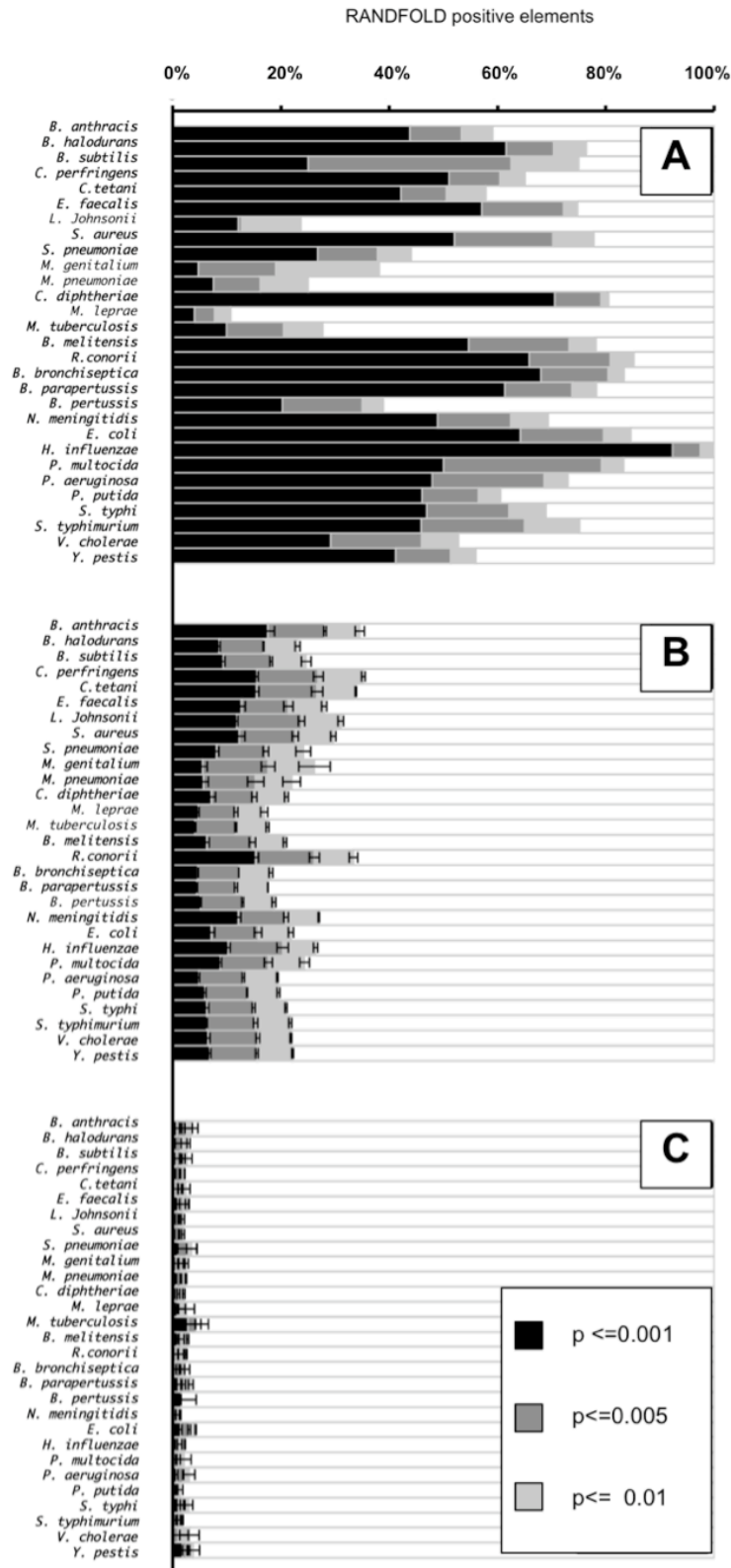


Figure 8. Randfold analysis

Fraction of sequence elements positive to RANDFOLD test. RANDFOLD test was run onto groups of clustered SLs (panel A), total SLs (panel B) and random sequences (panel C) from the 29 genomes listed in Table 3. The fraction of elements scoring positive with the indicated probability is diagrammed. Standard deviation bars are shown in panels B and C.

Finding relations between clusters

In order to detect possible relationships between clusters, various grouping procedures were attempted, based on sequence similarity, strand reciprocity and position on the genome. The results, reported in Table 4, allowed to further combine the initial 523 clusters into a smaller number.

A first grouping strategy was aimed to pull together clusters whose elements are similar at sequence level, as the first clustering procedure was very stringent and elements of the same type were likely to be separated in different clusters. The procedure involved re-clustering SCRs by reusing the same BLAST and MCL tools, under less stringent conditions. This analysis reduced the 523 clusters to 301, most of them characterized by a larger number of elements, as shown in column ‘sequence’ of Table 4. Within each new cluster, overlapping SCRs were further combined as described above, to produce even larger non-overlapping regions.

A second strategy was used to verify the presence of clusters whose members are similar but located on opposite strands, i.e. are reverse complement. The idea is based on the evidence that the ability to form SLS is generally shared by the two complementary strands of a given DNA sequence, except for sequences where G-U pairing is essential to form a stem-loop satisfying the minimum requirements. For this reason, a number of clusters are likely to be composed of elements from the opposite strands of the same genomic region. Again the BLAST-MCL procedure was used to detect this kind of clusters, but this time allowing BLAST searches also on the complementary strand. About two thirds of the clusters could be paired in this way, thus the total number was reduced to 205 ‘unrelated’ clusters, as seen in column ‘strand’ of Table 4.

The third strategy was used to group clusters whose members represent different parts of a larger DNA repeat. To this aim, the genomic position of all members of each cluster have been compared in order to find clusters with most elements overlapping or located at short

distance (< 150 bp). Once detected, these clusters were joined within one group. This led to a further reduction to 137 cluster groups reported in column 'location' of Table 4.

Finally, the resulting set was analyzed by searching again for ISs and repeated structured RNAs such as tRNA and rRNA, trying to identify sequences missed during the first filtering. SCRs of each cluster were compared with the IS sequences collected in the ISfinder database [Siguier et al. 2006] by using BLAST, in order to remove clusters whose members match with ISs not described at the time of the initial selection. Clusters related to rRNA and tRNA were removed by evaluating the genomic localization of their elements respect to those of genes encoding stable RNAs. These tests revealed that 28 cluster groups are composed of sequences related with Insertion Sequences, mostly not known at the time of the initial filtering, and 11 cluster groups were made by sequence elements contained within rRNA precursors. These 39 cluster groups, reported in the columns 'IS' and 'rRNA' of Table 4, have been tagged and excluded in further analysis.

The whole procedure above described led to the selection of 98 candidate SLS-containing repeated DNA families.

Species	Clusters	Grouped by			Located within	
		sequence	strand	location	IS	rRNA
<i>B. anthracis</i>	4	3	2	2		
<i>B. halodurans</i>	6	6	4	3		1
<i>B. subtilis</i>	2	2	1	1		1
<i>C. perfringens</i>	6	2	1	1		
<i>C. tetani</i>	14	13	10	6	3	
<i>E. faecalis</i>	7	5	3	3	1	
<i>L. johnsonii</i>	3	3	2	2	1	
<i>S. aureus</i>	11	7	5	4		
<i>S. pneumoniae</i>	28	22	13	9	6	
<i>M. genitalium</i>	1	1	1	1		
<i>M. pneumoniae</i>	20	20	18	12		
<i>C. diphtheriae</i>	9	7	5	4	1	
<i>M. leprae</i>	29	18	11	5		
<i>M. tuberculosis</i>	59	36	21	15	3	
<i>B. melitensis</i>	11	7	5	4		
<i>R. conorii</i>	19	6	4	4		
<i>B. bronchiseptica</i>	26	8	5	4		
<i>B. parapertussis</i>	30	16	10	5	4	
<i>B. pertussis</i>	52	28	16	4	3	
<i>N. meningitidis</i>	44	9	7	6		
<i>E. coli</i>	12	8	6	6		2
<i>H. influenzae</i>	3	1	1	1		
<i>P. multocida</i>	1	1	1	1		
<i>P. aeruginosa</i>	9	5	4	4		
<i>P. putida</i>	75	35	26	14	4	2
<i>S. typhi</i>	8	4	3	3		2
<i>S. typhimurium</i>	7	6	4	4		1
<i>V. cholerae</i>	7	7	5	4		2
<i>Y. pestis</i>	20	15	11	5	2	
Total	523	301	205	137	28	11

Table 4. Regrouping of SLS clusters

Clusters reported in Table 3 were tested for sequence similarity, strand reciprocity and relative genomic position of their elements, and grouped accordingly. The number of clustered groups is reported in columns marked “Grouped by”. The number of groups, whose elements are part of ISs or rRNA genes, is shown in the last two columns.

Expanding detected repeated families by using Hidden Markov Model

The procedures described above are not able to check whether cluster members are part of larger DNA repeats whose boundaries do not coincide with those of SLSs. Moreover, it is also possible that other genomic sequences similar to members of detected family may exist even if not containing any SLS.

For these reasons, a combined iterative procedure, based on Hidden Markov Model (HMM) genome searches, was developed and applied to each identified family, aimed to identify the complete set of family members. HMM is a statistical model in which the system being modelled is assumed to be a stochastic process with unknown parameters (Markov process). Hidden parameters are estimated starting from a known set of data and are then used to perform further analysis, such as pattern recognition. A sequence alignment can be described by a HMM that can in turn be used to detect new sequences able to fit to it.

In this procedure, a HMM is built starting from the alignment of all family members and used to scan the parental genome to detect similar sequences. Detected sequences are then aligned to the model and alignments are extended by attaching neighbouring sequences, in order to define larger models, when possible. Multiple cycles of alignment, elongation, model building and genome search were performed until the borders of the repeated sequence were reached (see Methods). The entire procedure is schematically represented in figure 9 and an example of results obtained from the elongation process is shown in figure 10.

At the end of this procedure, if two or more models identify identical sequences on the genome, they were considered equal and the corresponding families were fused, leading at the final identification of 92 models, which define the families reported in Table 5, together with the length of the model and the number of detected sequences, both covering

the entire model or part of it. 67 models range in size between 31 and 200 bp, while the rest are larger than that, although only two extend over 1 Kb.

Since some of the repeated families have already been described and sometimes even analyzed in depth in the literature, consensus sequences for DNA repeats described in literature have been used to scan members of detected families by BLAST. This comparison reveals that 25 families are already known and correspond to essentially all previously identified SLS containing families. For each of them, size and copy number are reported in Table 5, along with the corresponding values derived from literature data [Mazzone et al 2001, De Gregorio et al. 2005, De Gregorio et al. 2006, Okstad et al. 2004, Martin et al. 1992, Oggioni et al. 1999, RicBase Rickettsia genome database, Cole et al. 2001, Parkhill et al. 2000, Bachellier et al. 1999, Sharples et al. 1990, Aranda-Olmedo et al. 2002].

The remaining 67 families are not described as such in literature. Their sizes range from 31 bases to over 2 kbs for a number of elements varying between 9 and 164. Nine of these families (Bhal-2, Clot-2, Clot-3, Myt-5 Sal-2, Myt-11, Nem-4, Pam-1, Hin-1) contain little previously described DNA sequence motifs, such as CRISPR [Godde et al. 2006], MIRU [Supply et al. 2000] and DUS [Davidsen et al. 2004]. The combination of two or more specific elements, matching these motifs, generates larger, SLS containing, repeated sequences not previously described. Sixteen families are made up of sequences contained within larger sequence blocks, either coding for abundant protein motifs or located within larger, ill-defined redundant intergenic sequences. 42 families appear to be unrelated to previously described sequence elements.

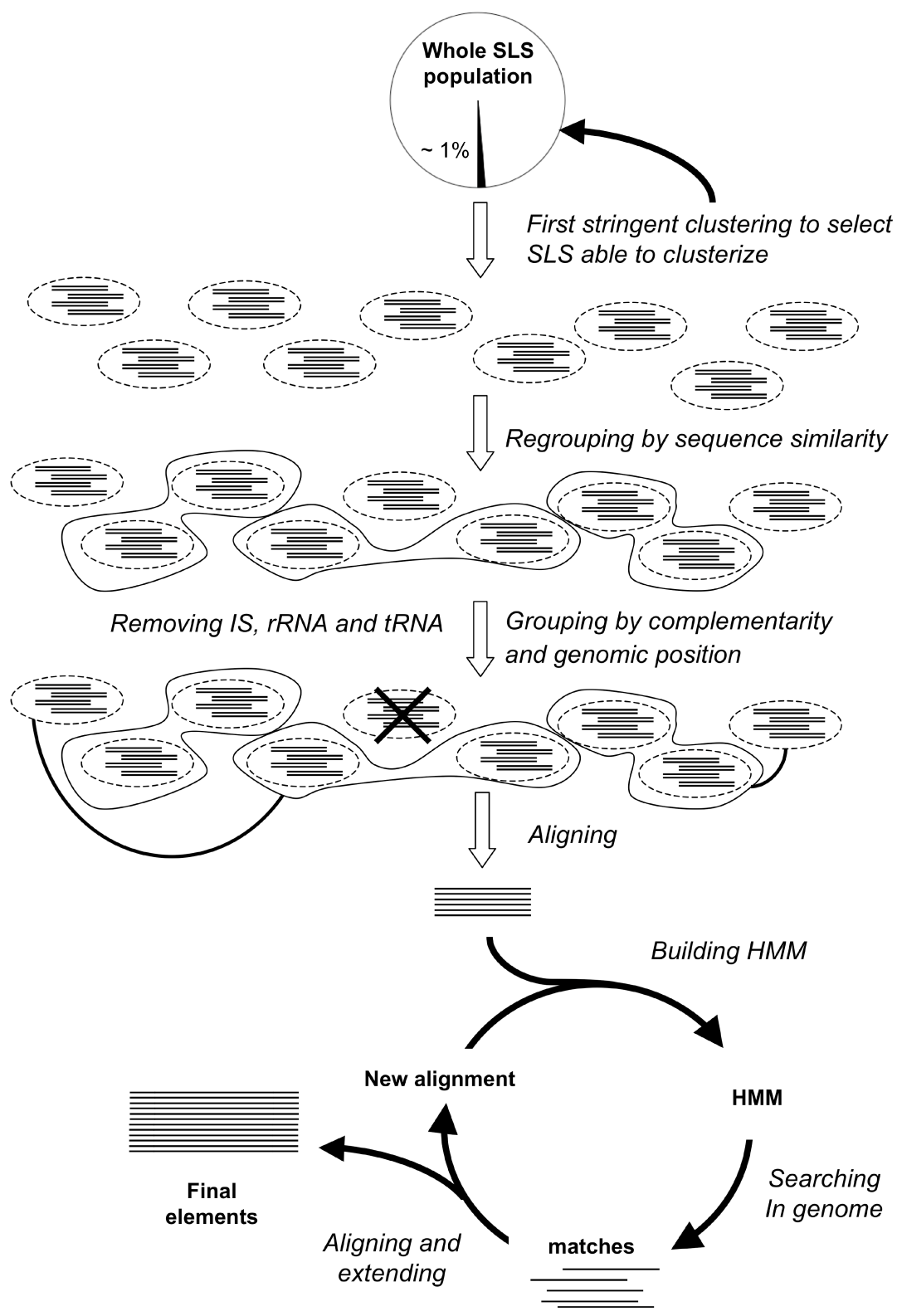


Figure 9. SLS pipeline flowchart

Schematic representation of the procedure used to detect repeated sequences containing SLSs.

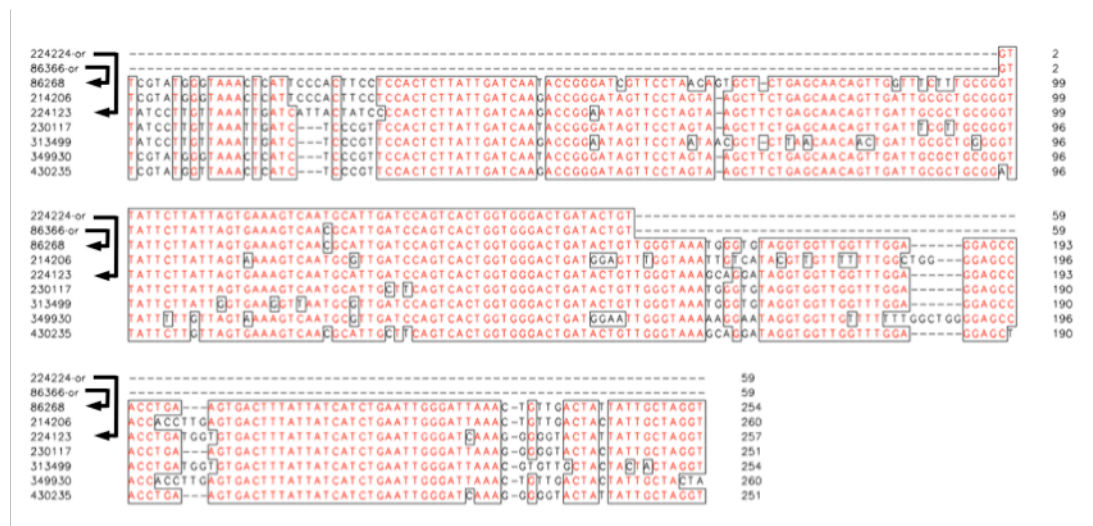


Figure 10. Elongation process

Two sequences of *M. genitalium* Myg-1 family detected by the clustering procedure are aligned with those obtained by the elongation process described in Methods. Arrows indicate the same sequences before and after the process.

Species	Family	This work		Literature			Type	Notes
		size	copies	size	copies	ref.		
<i>B. anthracis</i>	Bant-1	72	104 (29)				I	
	Bcr1	167	31 (21)	147	12	[A]	I	
<i>B. halodurans</i>	Bhal-1	74	36 (32)				I	
	Bhal-2	76	50 (41)				I	contains CRISPR repeats
<i>C. perfringens</i>	Clop-1	93	44 (28)				I	
<i>C. tetani</i>	Clot-1	74	19 (16)				I	
	Clot-2	31	34 (32)					contains CRISPR repeats
	Clot-3	90	24 (17)				I	contains CRISPR repeats
<i>E. faecalis</i>	Efa-1	163	65 (18)				I	
	Efa-2	292	11 (9)				G	
<i>L. johnsonii</i>	Lac-1	231	34 (6)				G	
<i>S. aureus</i>	Sta-1	105	25 (25)				I	
	Sta-2	460	9 (8)				S	
	Sta-3	136	24 (15)				I	
	Sta-4	99	46 (27)				I	
<i>S. pneumoniae</i>	BOX	84	205(105)	100-200	127	[B]	I	
	RUP	63	110 (99)	108	54	[C]	I	
	Stre-1	45	241(225)				G	
<i>B. melithensis</i>	Bru-RS	118	222 (69)	103-105	35-40	[D]	I	
<i>R. conorii</i>	Rpe-4	100	97 (74)	95	94	[E]	I	
	Rpe-5	115	45 (35)	115	55	[E]	I	
	Rpe-6	108	123 (74)	136	168	[E]		
	Rpe-7	123	186 144)	99	223	[E]		
<i>M. genitalium</i>	Myg-1	259	10 (7)				I	
<i>M. pneumoniae</i>								
	Myp-1	143	25 (18)				G	part of REPMP1

Species	Family	This work size copies	Literature size copies	ref.	Type	Notes
						repeat
	Myp-2	158	42 (16)		G	part of REPMP4 repeat
	Myp-3	558	11 (8)		G	part of REPMP5 repeat
	Myp-4	364	8 (7)		G	part of REPMP5 repeat
	Myp-5	426	8 (8)		G	part of REPMP5 repeat
	Myp-6	468	11 (11)		G	part of REPMP2/3 repeat
	Myp-8	674	9 (9)		G	part of REPMP2/3 repeat
	Myp-9	226	9 (9)		G	part of REPMP2/3 repeat
	Myp-10	330	12 (12)		G	part of REPMP2/3 repeat
	Myp-7	131	42 (22)		G	
<i>C. diphtheriae</i>	Cod-1	140	17 (16)		I	
	Cod-2	32	43 (39)		G	
	Cod-3	170	23 (20)			
	Cod-5	74	35 (29)		I	
<i>M. tuberculosis</i>	Myt-1	72	75 (70)			
	Myt-2	115	769(223)		G	located within PE genes
	Myt-3	81	81 (77)		G	located within PE genes
	Myt-4	83	196 (68)		G	located within PE genes
	Myt-5	71	41 (2)		G	contains CRISPR repeats
	Myt-7	136	278 (68)		G	located within PE genes
	Myt-8	92	33 (25)			
	Myt-9	67	53 (15)			
	Myt-10	154	62 (59)		G	located within PE genes
	Myt-11	65	56 (21)			contains MIRU repeats
<i>M. leprae</i>	REPLEP	740	29 (9)	400-880	15 [F]	I
	RLEP	641	38 (30)	601-1075	37 [F]	S
	My1-1	371	7 (4)			S
	My1-2	1979	9 (7)			S
<i>B. bronchiseptica</i>	Bor-1	117	196 (92)			I
	Bor-2	167	17 (6)			I
	Bor-3	134	34 (32)			G
	Bor-4	81	164(114)			G
	Bor-5	112	135(101)			G
	Bor-6	147	37 (31)			G
<i>B. pertussis</i>	Bor-1	93	128 (78)			I
<i>N. meningitidis</i>	ATR	206	14 (9)	183	13 [G]	I
	Nem-2	341	11 (7)			
	Nem-3	127	10 (9)			G
	Nem-4	36	412(362)			I
	dRS3	33	755(708)	20	770 [G]	I
	NEMIS	46	262 (81)	106-158	250 [H]	I
	Rep2	65	22 (18)	59-154	26 [G]	I
<i>P. multocida</i>	Pam-1	155	12 (12)			S
<i>E. coli</i>	BoxC	50	22 (20)	56	32 [I]	
	Eco-1	734	9 (7)			G

Species	Family	This work size copies	Literature size copies	ref.	Type	Notes
	ERIC	140	19 (19)	127	21 [J]	S
	PU-BIME	108	301(199)	40	485 [I]	
<i>H. influenzae</i>	Hin-1	31	53 (51)			I contains DUS repeats
<i>P. aeruginosa</i>	Pae-1	84	133 (61)			I
	Pae-2	287	65 (24)			G
	Pae-3	220	16 (13)			G
	Pae-4	52	41 (35)			
<i>P. putida</i>	Ppu-1	617	39 (28)			I
	Ppu-2	2056	10 (8)			S
	Ppu-3	251	27 (23)			G
	Ppu-4	81	41 (24)			I
	Ppu-9	124	57 (31)			I
	REP	39	588(496)	30	804 [K]	I
<i>S. typhi</i>	PU-BIME	43	146(126)	40	100 [I]	I
	PU-BIME*	80	59 (37)	40	>100 [I]	
<i>S. typhimurium</i>	PU-BIME	78	142 (94)	40	82 [I]	
	Sal-1	115	27 (17)			I
	Sal-2	120	33 (3)			G contains CRISPR repeats
<i>V. cholerae</i>	ERIC	103	97 (66)	127	80 [I]	I
	Vic-1	184	14 (1)			I
<i>Y. pestis</i>	ERIC	115	241(128)	69-127	167 [L]	I
	YPAL	168	101 (68)	169	30 [M]	I
	YPAL*	136	26 (13)	130	10 [M]	I

Table 5. Families of SLS containing repeated sequences.

The final set of 92 families of repeated sequences is reported, grouped by species. For each family, the length of the model and the number of sequences fitting the model are given. The number of complete sequences, i.e. covering the model from end to end, is reported in parenthesis. Previously described sequence families have been named in column “Family”, according to the current literature; for each of them, the number and typical size of its members are also provided, together with references indicated by letters: Okstad et al. 2004 [a], Martin et al. 1992 [b], Oggioni et al. 1999 [c], Halling et al. 1994 [d], RicBase [e], Cole et al. 2001 [f], Parkhill et al. 2000 [g], Mazzone et al. 2001 [h], Bachellier et al. 1999 [i], Sharples et al. 1990 [j], Aranda-Olmedo et al. 2002 [k], De Gregorio et al. 2005 [l] and 2006 [m]. For novel families, a systematic name was built by fusing a shortened species name to a progressive number. In the column “type”, I, G and S indicate the prevalent genomic location of the members of each families within intergenic, genic or border-spanning sequences. For some families, small previously described sequence motifs contribute to the formation of a substantially larger model; for others, their members are frequently located within larger previously described sequences. In both cases, a note is reported in the rightmost column.

Secondary structure analyses

Members of detected families were tested for their ability to share a common stable secondary structure by using three different approaches:

- 1) RNAz [Washietl et al 2005] was used to check for the presence of a conserved secondary structure within a family by analyzing an alignment of six representative sequences to their HMM (column “conserved structure” in table 6);
- 2) The presence of aligned SLSs was compared with the structure predicted by RNAz and agreement between them was evaluated (column “conserved SLS position” in table 6);
- 3) The probability of non-random folding for SLSs contained within each family was calculated by using RANDFOLD [Bonnet et al 2004] (column “SLS folding aptitude” in table 6).

Only families with either a predicted conserved secondary structure or aligned SLSs are reported in Table 6. 57 out of 92 families are predicted to have a conserved secondary structure by RNAz. For most (47) of them, marked as “s”, the predicted structure contains a stem-loop compatible with the original search. In all except for Cod-2, the position of the originally found SLSs is in agreement with the structure predicted by RNAz. Analyzing these SLSs by RNADFOLD revealed that 36 of the 47 families have most members with very stable SLSs ($P \leq 0.005$).

For ten of the 57 putative structured families, indicated by “c”, a complex common structure is predicted by RNAz, not including a stem-loop compatible with the original search. Most of them do not feature aligned SLSs. Only three families, *L. johnsonii* Lac-1, *M. leprae* REPLEP and *E. coli* BoxC, show discrepancies between aligned SLSs and stem-loop structures predicted by RNAz, suggesting alternative foldings.

RNAz is unable to predict a common structure for 35 of the 92 families: for most of these families (29 out of 35) no aligned SLSs are available, indicating the absence of common

secondary structures. Aligned SLSs are present in 6 families, *M. genitalium* Myg-1, *M. pneumoniae* Myp-1 and Myp-4, *E. coli* Eco-1, *P. aeruginosa* Pae-3 and *R. conorii* RPE-6, which show no positive score at the RNAz test. All but RPE-6 showed aligned SLSs that feature a low folding aptitude, calculated by RANDFOLD (see Table 6).

Species	Family	P	Conserved structure	Conserved SLS position	SLS folding aptitude	Type
<i>B. anthracis</i>	Bcr1	0.99	s	+	+	I
<i>B. halodurans</i>	Bhal-1	0.98	s	+	++	I
	Bhal-2	0.99	c		-	I
<i>C. perfringens</i>	Clop-1	0.96	s	+	+	I
<i>C. tetani</i>	Clot-1	0.95	s	+	++	I
<i>E. faecalis</i>	Efa-1	0.85	s	+	+++	I
	Efa-2	1.00	s	+	-	G
<i>L. johnsonii</i>	Lac-1	0.97	c	+°	-	G
<i>S. aureus</i>	Sta-1	0.84	s	+	+++	I
	Sta-2	1.00	s	+	++	S
	Sta-3	0.97	s	+	+	I
<i>B. melithensis</i>	Bru-RS	0.98	s	+	+	I
<i>R. conorii</i>	Rpe-4	0.73	s	+	-	I
	Rpe-5	1.00	s	+	+	I
	Rpe-6	0.45	-	+°	+	
	Rpe-7	0.99	s	+	++	
<i>M. genitalium</i>	Myg-1	0.06	-	+°	-	I
<i>M. pneumoniae</i>	Myp-1	0.00	-	+°	-	G
	Myp-2	0.95	s	+	++	G
	Myp-3	0.89	s	+	-	G
	Myp-4	0.09	-	+°	-	G
	Myp-5	0.74	s	+	-	G
	Myp-6	0.55	c		-	G
	Myp-7	0.67	s	+	-	G
<i>C. diphtheriae</i>	Cod-1	0.97	s	+	+++	I
	Cod-2	0.98	s		-	G
	Cod-3	0.99	s	+	+++	
<i>M. tuberculosis</i>	Myt-1	0.74	s	+	+++	
	Myt-8	0.90	s	+	++	
<i>M. leprae</i>	REPLEP	1.00	c	+°	-	I
	RLEP	1.00	s	+	++	S
	Myl-1	0.61	s	+	++	S
	Myl-2	0.97	s	+	+	S
<i>B. bronchiseptica</i>	Bor-1	0.86	s	+	++	I
	Bor-2	1.00	s	+	-	I
<i>B. pertussis</i>	Bor-1	0.93	s	+	++	I
<i>N. meningitides</i>	ATR	1.00	s	+	-	I
	Nem-2	0.93	s	+	+	
	Nem-4	0.93	s	+	+++	I
	dRS3	0.98	c		-	I
	NEMIS	1.00	s	+	+	I
	Rep2	0.98	s	+	+	I
<i>P. multocida</i>	Pam-1	0.96	s	+	+++	S
<i>E. coli</i>	BoxC	0.99	c	+°	-	
	Eco-1	0.18	-	+°	-	G
	ERIC	0.94	s	+	++	S
	PU-BIME	0.94	s	+	+	
<i>H. influenzae</i>	Hin-1	0.96	s	+	+	I
<i>P. aeruginosa</i>	Pae-1	0.97	s	+	++	I
	Pae-3	0.26	-	+°	-	G
	Pae-4	0.93	s	+	++	
<i>P. putida</i>	Ppu-1	0.97	s	+	+	I
	Ppu-2	1.00	s	+	+++	S
	Ppu-4	0.95	s	+	-	I
	Ppu-9	0.54	s	+	-	I
<i>S. typhi</i>	PU-BIME	0.97	c		-	I
	PU*-BIME	0.98	s	+	-	

	PU-BIME	0.98	s	+	-	
<i>S. typhimurium</i>	Sal-1	0.94	c		-	I
	Sal-2	1.00	c		-	G
	ERIC	0.90	s	+	-	I
<i>Y. pestis</i>	YPAL	1.00	s	+	+++	I
	YPAL*	0.96	c		-	I

Table 6. Secondary structure prediction analysis of families

The ability to form a consensus secondary structure was evaluated by RNAz: the prediction scores are reported in column “P” for each family. The type of predicted structure is indicated in column “conserved structure”, where “s” indicates a stem-loop based structure, while “c” indicates a more complex structure, where a stem-loop compatible with the original search is not present. For each family, the aligned localization of the original SLSs is indicated by ‘+’ in column “conserved SLS position”; when SLS alignment is not in agreement with the RNAz prediction, a ‘°’ is added to the ‘+’ symbol. The column marked “SLS folding aptitude” reports the behavior of family elements in the RANDFOLD test: the number of ‘+’ symbols describes the percent of positive elements (‘+++’ if 90% or above; ‘++’ if 70-90%; ‘+’ if 50-70%; ‘-’ if less than 50%). The localization of family members, as already described in Table 5, is also reported in the last column.

Genomic localization of detected families

Most members of the already described families are located within intergenic regions. For this reason, genomic localization of the identified families was analyzed and families are classified according to the position of the vast majority of their members, relative to annotated coding sequences (see Table 5 column “type”). 41 families are mostly intergenic (I), 30 genic (G) and 7 tend to span the borders between coding and non-coding sequences, and are therefore indicated as border spanning (S). 14 families have no clear predominance of genic or intergenic sequences, and, for this reason, were not assigned to a class. Genomic localizations are also reported in Table 6 for families that are predicted to fold in a secondary structure

For all families, genomic localization, correlated with the predicted ability of the family members to fold into a common, stable secondary structure, are summarized in Table 7. Most “intergenic” families show a predicted secondary structure (31 out of 41), in contrast to “genic” ones, that are predominantly not structured. In particular, only 9 out of 30 genic

families are predicted by RNAz to be structured and only 5 of them also have a supporting SLSs alignment. Border spanning and unclassified sequence families feature a predicted secondary structure with frequencies similar to intergenic ones.

Genomic location	Sec. Struct. +		Sec. Struct. -		Total
	SLS +	SLS -	SLS +	SLS -	
Genic	5	4	4	17	30
Border spanning	7	0	0	0	7
Intergenic	25	6	1	9	41
Others	9	1	1	3	14
Total	46	11	6	29	92

Table 7. Structural properties of the SLS families in relation to genomic location

Columns under “Sec. Struct. +/-” report the number of families, characterized by the presence or absence of a conserved secondary structure predicted by RNAz; the labels “SLS +/-” indicate the presence or absence of aligned SLSs; “Total” is the sum of rows or columns.

Characterization of specific families

The described procedure schematically represented in figure 9 led to the identification of a large number of families of repeated bacterial sequences, some already known, other not previously described. For many of them, a number of tests showed the potential folding of the majority of their members into a shared secondary structure. Four examples of such families are reported in figures 11, 12, 13, 14, 15, 16 and 17 where the predicted secondary structure is shown along with the aligned, originally found, SLSs. One of them, the ERIC family from *E. coli* (see Figure 11), was previously described, while the other three are new ones. ERIC elements, as anticipated from literature reports [Bachelier et al, Sharples et al 1990], are predicted to fold into a single, long stem-loop structure. Sta-1 family (Figure 12) is composed of sequences able to fold into a simple, shorter SLS. Pae-1 and Efa-1 families (Figures 13 and 14) feature more complex structures, composed of a pair of

adjacent SLSs. The structures predicted for these four families may be predicted on both strands, with complementary sequences generally, but not necessarily, folding into corresponding stems. For Pae-1, the prediction of different structures on the two strands indicates the likely presence of multiple foldings of comparable stability, which, on each strand, are alternatively selected as the best one, because of minor base pair differences.

Two families, *M. tuberculosis* Myt-1 and *P. auruginosa* Pae-4, share a predicted secondary structure symmetrically located on both strands. Their members are frequently found within intergenic regions located between convergently transcribed genes, a position compatible with a putative function as bidirectional terminators, as schematically represented in figure 18. For some of the identified families, secondary structure predictions, although supported by high RNAz scores, are not consistent with the originally found SLSs. Generally this stems from the prediction, by RNAz, of structures not including SLSs fitting with the original SLS definition. PU-BIME and dRS3, shown in figures 15 and 16, are examples of such families: in PU-BIME the stem includes a five base internal loop, while in dRS3 the 8 bp stem is too short. Both cases are not compatible with the original search (see Methods). Finally, for about one third of the 92 identified families, it is unlikely that the RNA secondary structure play a relevant role, as shown by the absence of either a common predicted structure or alignment of originally found SLSs. An example of such families is Myt-10, reported in figure 17.

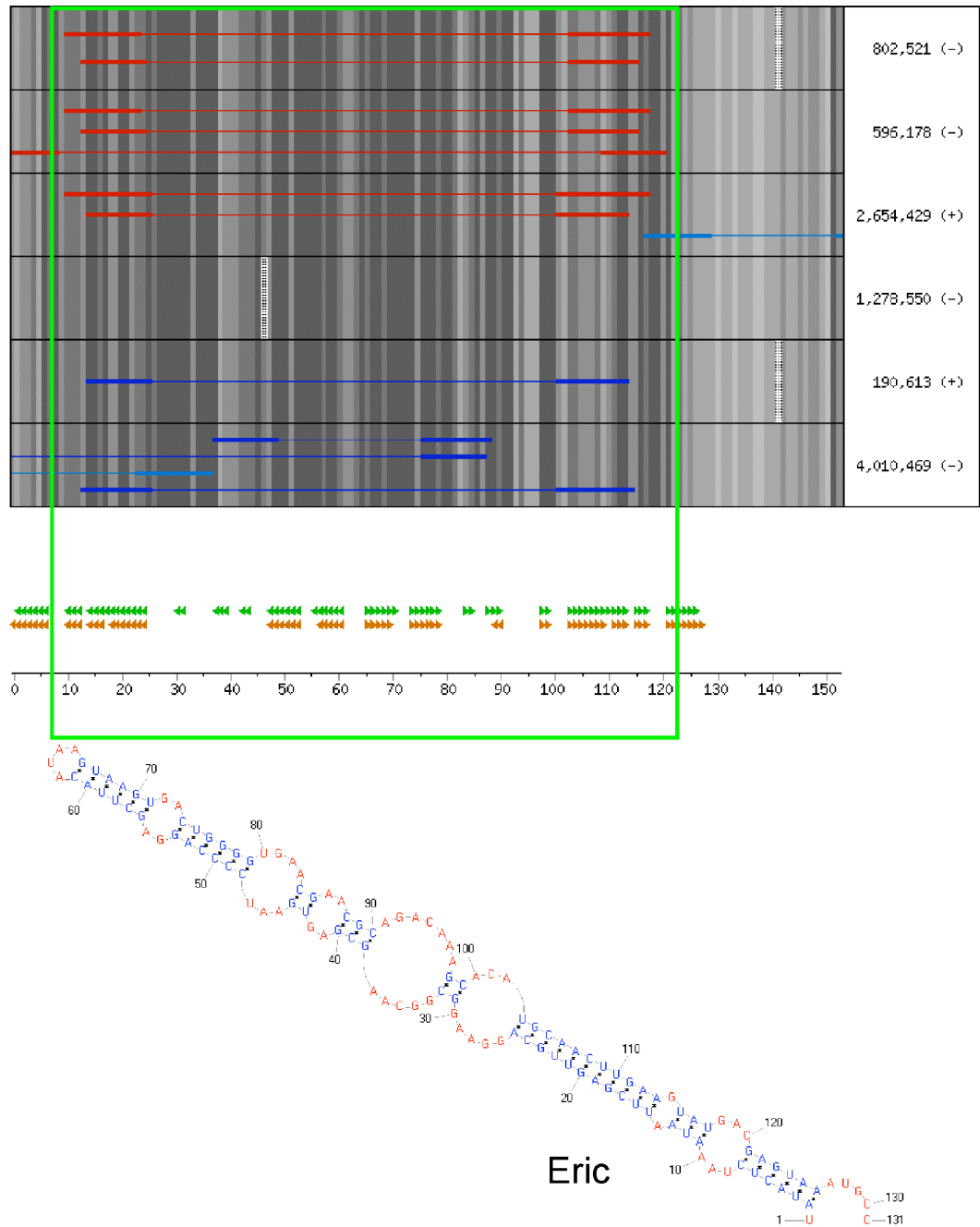


Figure 11. ERIC family (*E. Coli*)

A representative set of elements from the indicated family was aligned by using the HMM model as a guide. In each panel, one row corresponds to one family member (indicated on the right with its genomic position). Within each row, sequence conservation is indicated by increasing gray levels and gaps by dotted spaces; overlapping SLSs are reported as red and blue lines, the red ones indicating SLSs used to define the original HMM model for the family, the blue all the others. Darker colors indicate the SLS folding aptitude, i.e. positivity to RANDFOLD for $P \leq 0.005$. Common secondary structures, predicted by RNAz, are reported at the bottom, just above the ruler in nucleotides: green triangles indicate stems produced by pairing complementary regions on the same strand as the identified SLSs, while brown triangles indicate the same from the opposite strand. The boxed regions highlight areas where aligned SLSs and predicted structures are in agreement. If present, the graphic representation of the secondary structure predicted by RNAz was reported. Structure was made by using the by Pseudoviewer software.

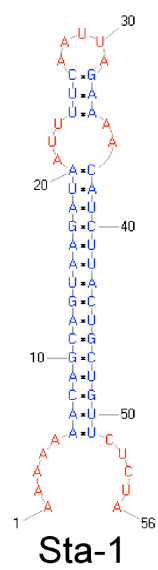
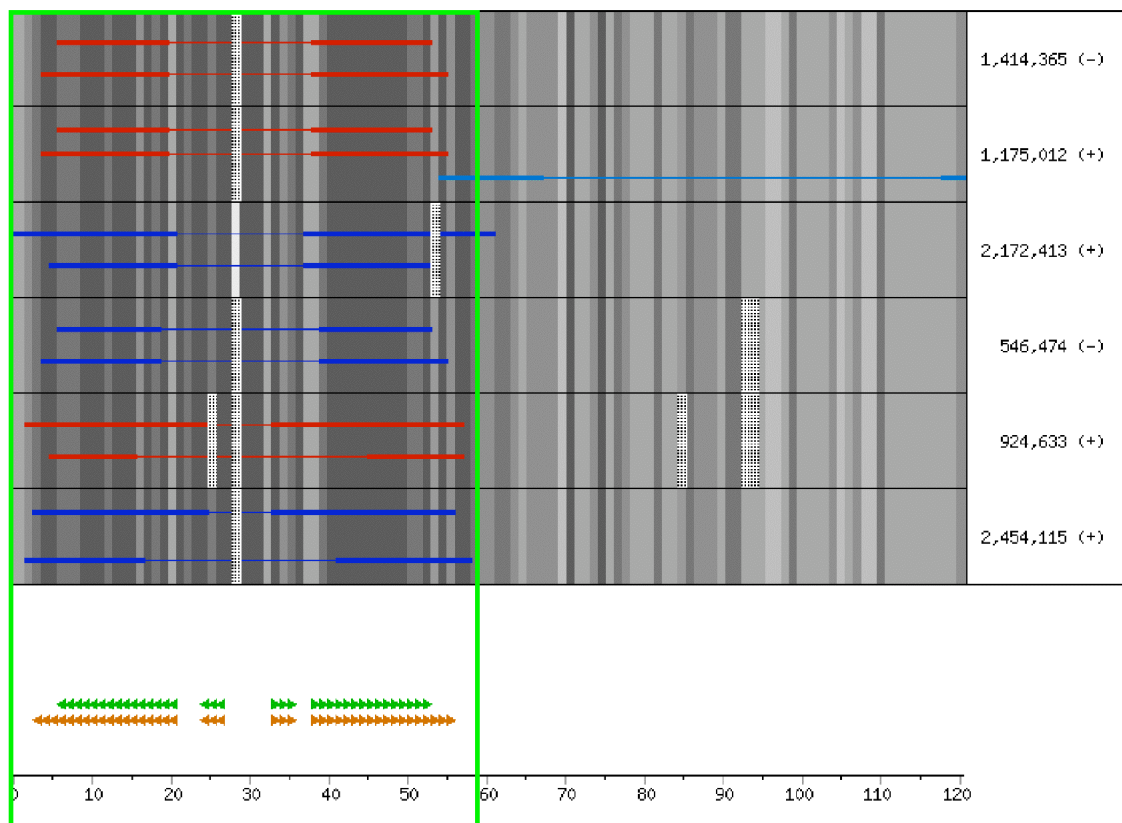


Figure 12. Sta-1 family (*S. aureus*)

The image description is given in figure 11.

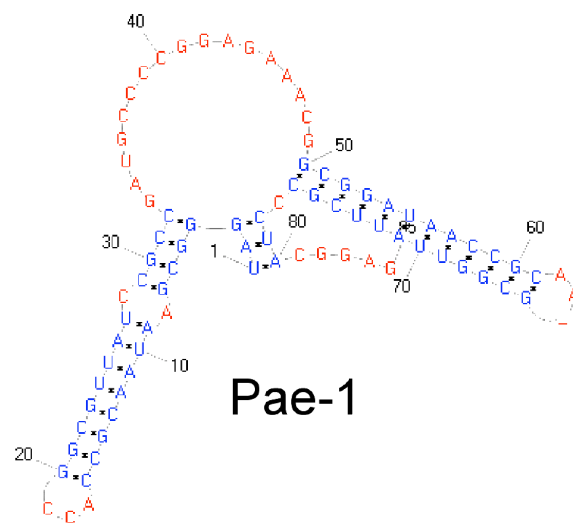
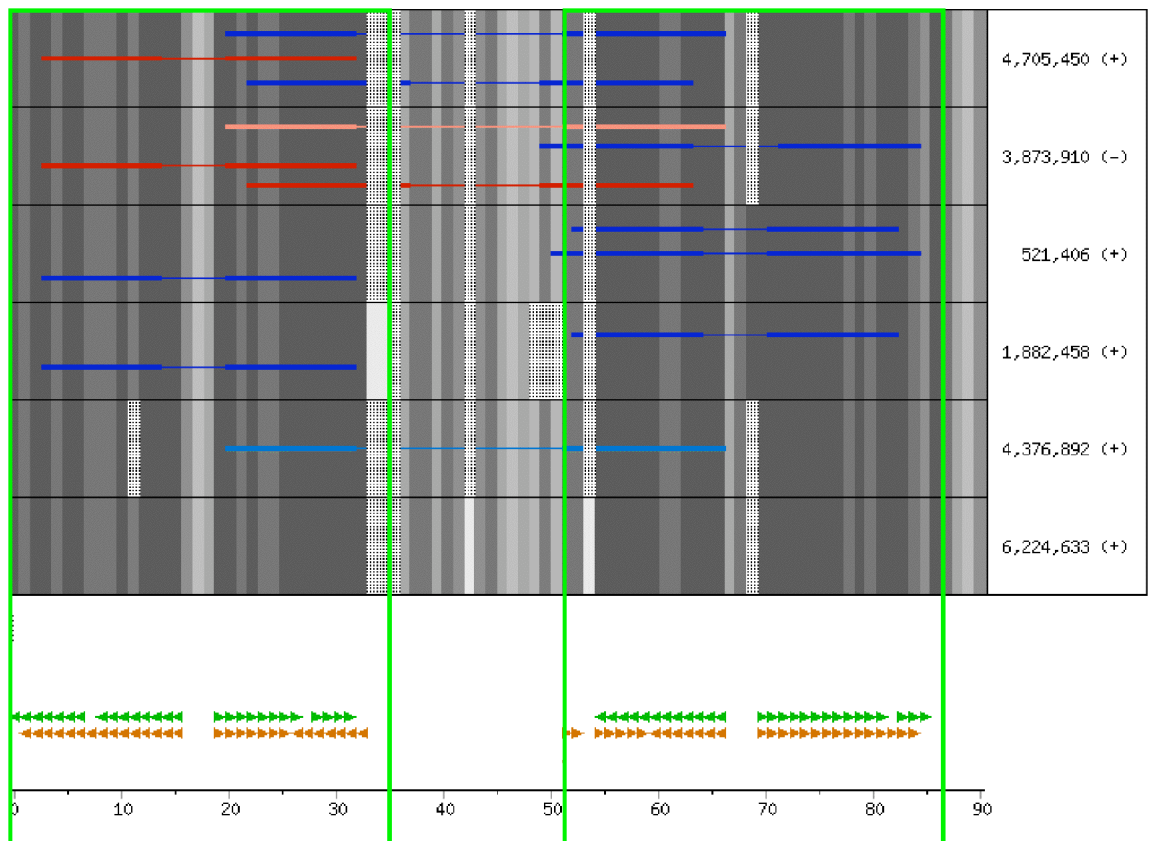


Figure 13. Pae-1 family (*P. auruginosa*)
The image description is given in figure 11.

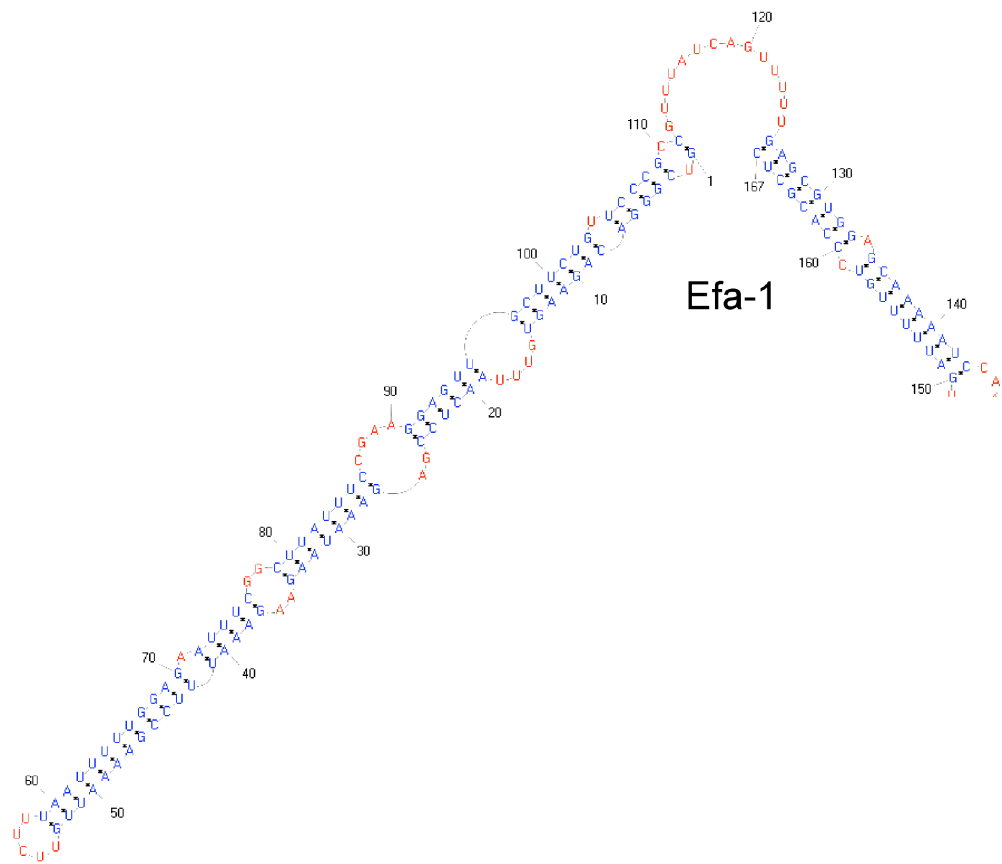
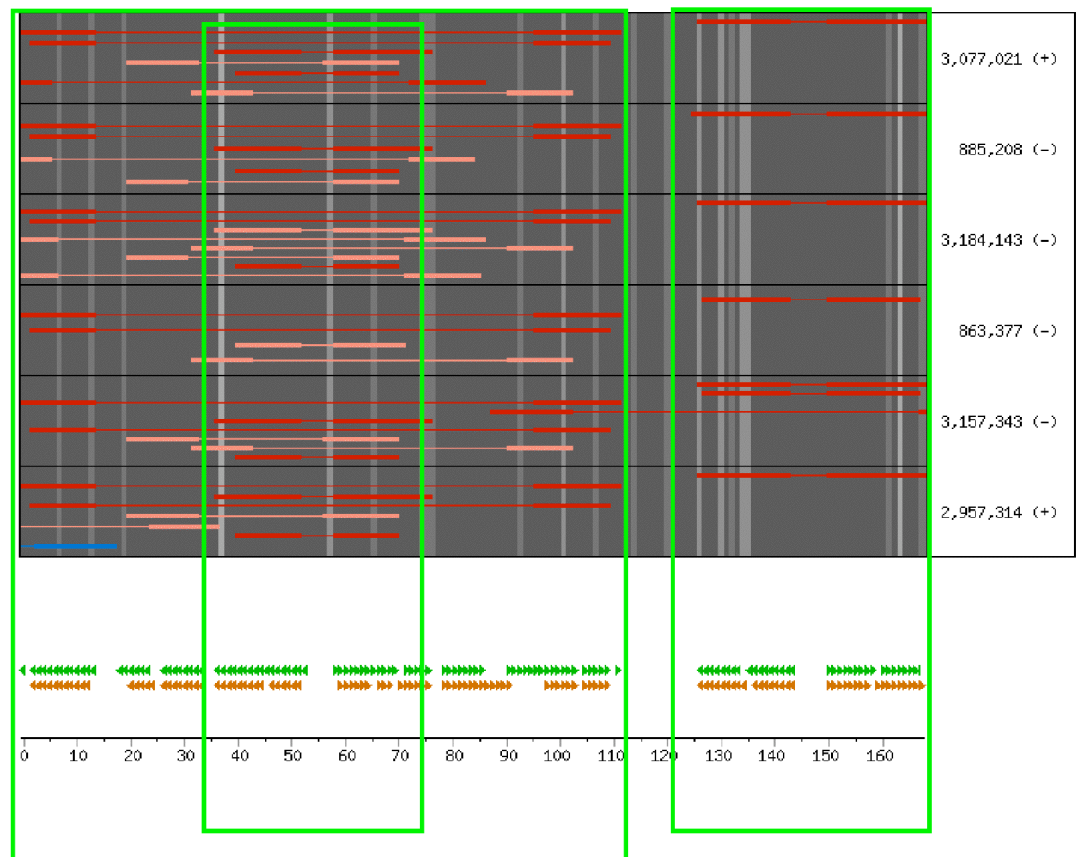
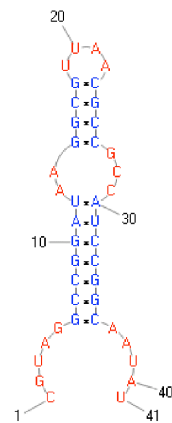
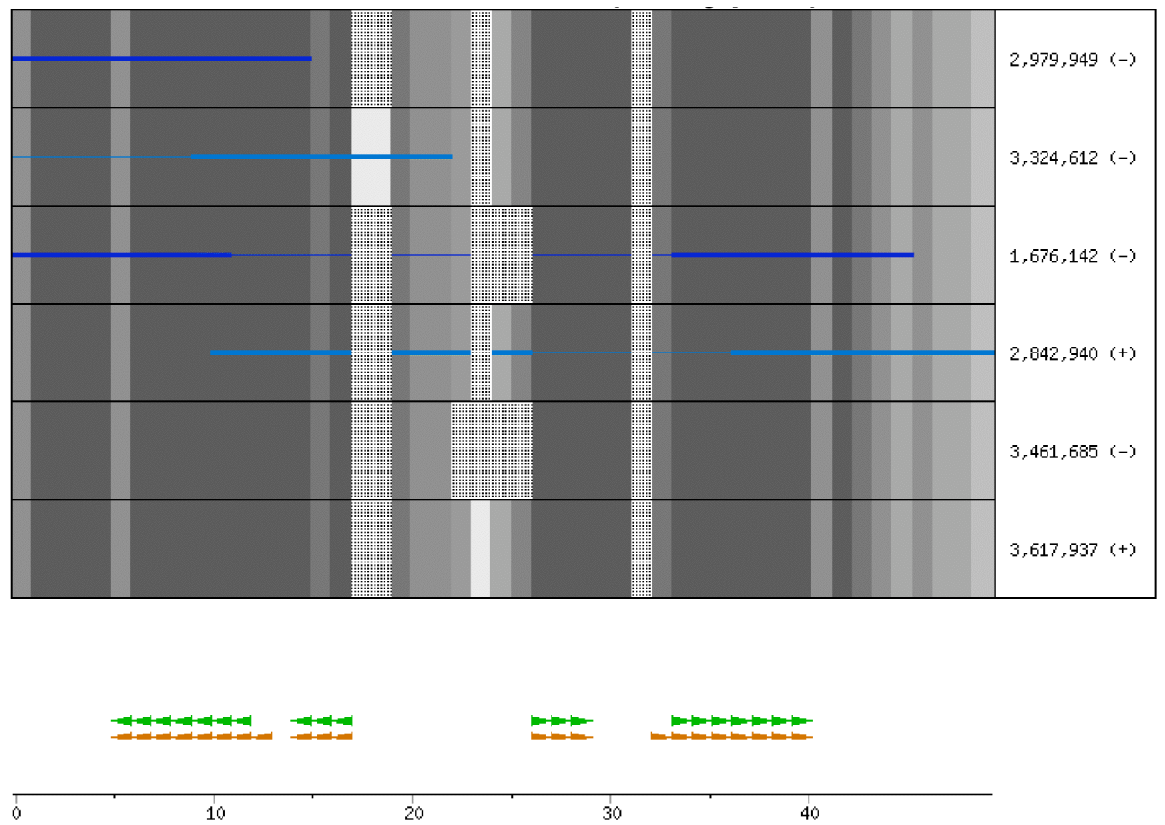


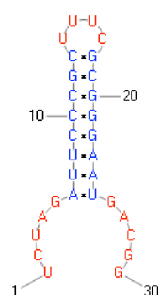
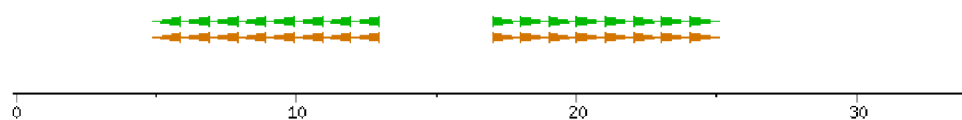
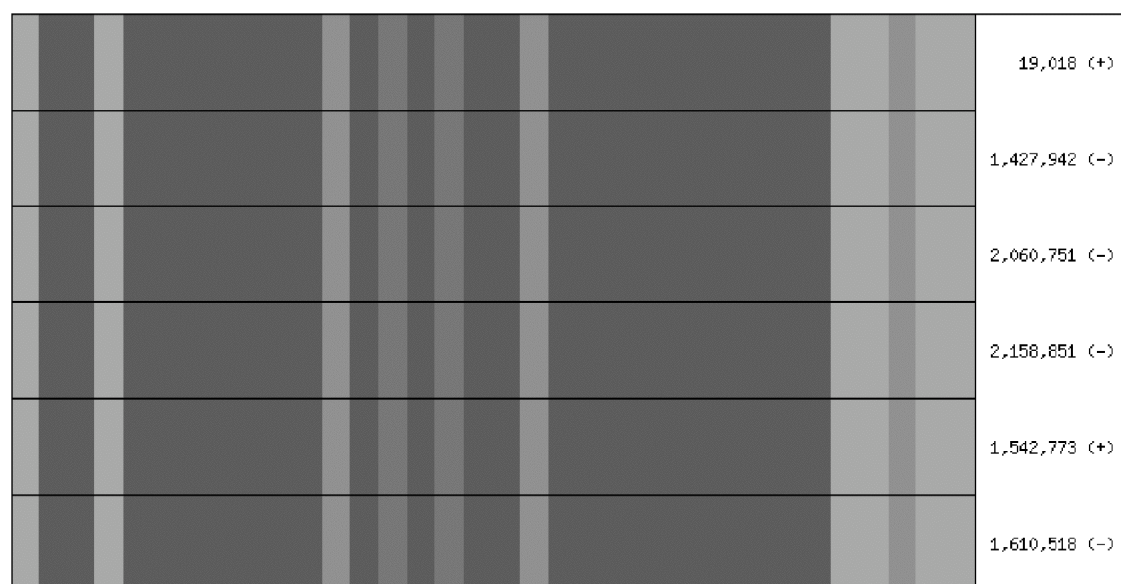
Figure 14. Efa-1 family (*E. fecalis*)

The image description is given in figure 11.



PU-BIME

Figure 15. Pu-BIME family (*S. typhi*)
The image description is given in figure 11.



dRS3

Figure 16. dRS3 family (*N. Meningitidis*)

The image description is given in figure 11.

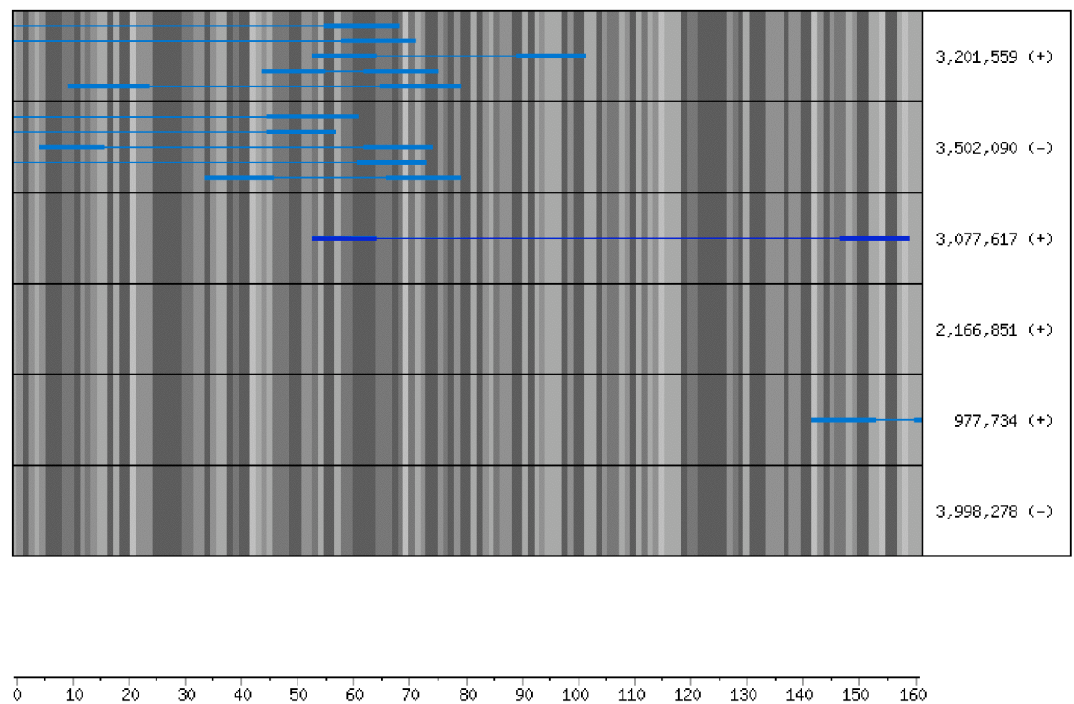


Figure 17. Myt-10 family (*M. tuberculosis*)

The image description is given in figure 11.

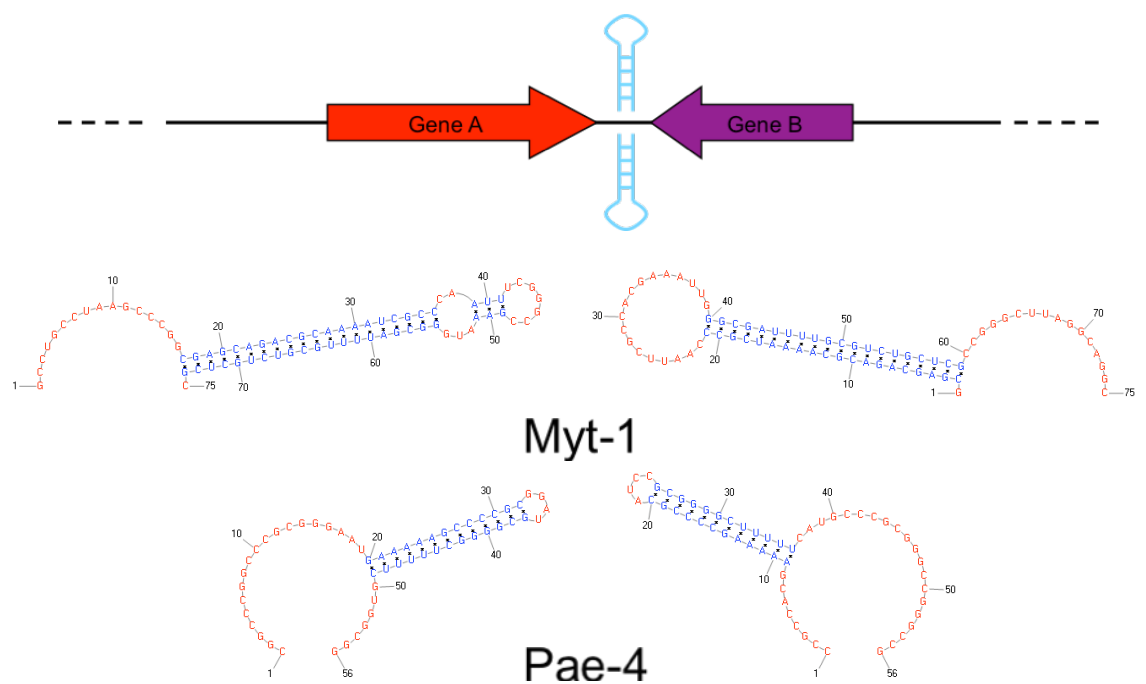


Figure 18. Myt-1 (*M. tuberculosis*) and Pae-4 (*P. aeruginosa*) families

Reported families are analyzed by RNAz on both strands. Predicted secondary structures are reported.

Discussion

Many new classes of functional elements have been identified in eukaryotes within non-coding genomic sequences and understanding their role pointed to relevant biological processes including development, control of proliferation and pathogenesis of diseases. Screening for secondary structure conservation, often in combination with comparative analysis, was used to detect families of functional RNAs such as miRNAs and snoRNAs. This approach is more difficult to use in the prokaryotic world because of the high plasticity of their genomes and the reduced amount of intergenic sequences. Still, in bacteria, SLSs are known to be essential in different aspects of gene expression and in regulation of biological pathways. Some of them are known to be involved in transcriptional attenuation and termination [Merino et al. 2005, Ermolaeva et al. 2000] and in regulation of mRNA stability [Higgins et al. 1988]. Others form cis-acting regulatory regions [Nudler et al. 2004] or participate to the formation of the catalytic site within enzymes such as RNase P [Kazantsev et al. 2006]. In some organisms, such as *Listeria monocytogenes*, a SLS within the 3' UTR of a virulence gene is known to regulate invasion of mammalian cells [Johansson et al. 2002] by acting as a RNA thermosensor: at low temperature it prevents expression by masking the ribosome binding site, when the temperature rises over 37 degrees, its disruption allows translation of the virulence gene thus inducing host invasion.

Here an attempt is described to systematically detect structured sequence families by looking at conservation within a bacterial genome. This study originated from the observation made by Petrillo et al. [2006] that natural genomes contain more high stability SLSs than artificial sequences produced by randomly shuffling their original sequence. Even if a large fraction of SLSs are expected to be formed by chance, this unbalance suggested that some sequences, and particularly, those able to form stable structures, could be preserved by selective pressure, possibly being involved in specific biological function.

A systematic approach was used to identify and classify families of repeated sequences that share a common secondary structure. This screening was performed on 40 genomes of bacterial species representing the prokaryotic divisions that are mostly involved in diseases, by using a procedure based on clustering of genomic stretches able to fold in a stem loop structure by sequence similarity in order to select only the repeated SLSs. The clustering procedure selects a subset composed by 1% of initial SLS population detecting clusters composed by a least 7 non-overlapping sequences in 29 of 40 analyzed genomes. Interestingly, although the clustering procedure is based exclusively on sequence similarity the resulting clusters have been found to be composed by sequences whose potential to fold into a stable secondary structure is considerably higher, if compared with the initial population. The fraction of SLSs that can be grouped by sequence similarity ranges from a consistent 6% of *N. meningitides* to a small 0.1% of *B. subtilis* and *P. multocida*. Looking for the ability of clustered SLSs to fold into a reliable secondary structure reveals that only few genomes show a low fraction of stable SLSs. Since these genomes have a GC content varying from 31.6 of *Mycoplasma genitalium* to 65.5 of *Mycobacterium tuberculosis* it is likely that GC content does not affect these results. After the refinement steps that are described in result section, 137 groups of clusters have been identified. These groups have been pruned by removing the ones that have members falling within different copies of rRNA and tRNA precursor or within ISs escaped from the initial filtering. Sequences belonging to each group have been used to build Hidden Markov Models that then were used to scan the original genome to detect all the similar sequences, including those not containing any SLS. In this way are detected and recognized also repeated families that only incidentally included SLSs within some their members. The procedure allows also fusing some groups that are included within a very large repeat or with HMMs that identify the same sequences. Finally the resulting 92 families have been analyzed in detail for their ability to share a common secondary structure.

Since clustering was performed by only looking for sequence similarity it is possible, in principle, that some of the detected families contain different SLSs. Moreover the HMM procedure, by looking for primary structure, may also extend sequences over areas not containing SLSs. Within the families, 35 were indeed identified with no recognizable shared secondary structure. Interestingly the majority of members that compose these families are located within coding regions where the formation of secondary structures is expected to be limited by the translation machinery. Also few previously described families such as *S. pneumoniae* BOX and *P. putida* REP are predicted to have no common secondary structure notwithstanding they have members located within intergenic portions. This can be related to the fact that their putative structure is not compatible with the initial SLS definition.

Families predicted to share a common secondary structure

About two thirds of the identified families are predicted by RNAz to have a common secondary structure. Many previously well-characterized intergenic families, such as *E. coli* PU-BIME and ERIC repeats, fall within this group as well as families that are only reported as simple repeats and on which no experiments have been made to address their function. With only two exceptions, all the known families, for which a secondary structure was predicted or demonstrated, fall within structured families, and their sequence boundaries are mostly coincident with those reported in literature. Only *S. pneumoniae* RUP and the *R. conorii* RPE-6 repeats are not recognized as structured although they are correctly recognized as repeated families. For RUP family it is thinkable that absence of conserved structure is caused by the recognition of only a portion of repeat by the pipeline. In some cases, in fact, the HMM extension procedure was unable to extend the initially detected sequences to cover the entire repeat. In addition to *S. pneumoniae* RUP family also the *N. meningitidis* NEMIS is only partially identified. In particular for RUP repeat

only 63 out of 108 bases were detected, while for NEMIS only the central 46 bp core common to both partial 108 and 158 bases repeats described in the work of Mazzone et. al [Mazzone et al. 2001] was identified.

Known and novel families

Although enterobacteria have the best characterized genomes a new repeated family that we named Eco-1 has been identified within the *Escherichia coli* genome. This family unlike the ones that are already reported in literature seems to have no predicted common secondary structure. On the other hand the well studied PU-BIME, ERIC and BoxC families are correctly predicted to be structured. Some of these families have been already described in different copy number within related bacterial genomes. This procedure identified the PU-BIME repeats also in *S. typhi* and in *S. typhimurium*. Our procedure identifies two variants of PU-BIME in *S. typhi*: a full-size and a shorter one while *S. typhimurium* seems to contain only the longer one. All these families share a secondary structure even if the full-size *S. typhi* PU-BIME shows a more complex situation. In *S. typhimurium* also two novel families, Sal-1 and Sal-2 have been detected able to share a conserved secondary structure. ERIC families has been detected in *E. coli*, *Y. pestis* and *V. cholerae* and this results are in according to the works of De Gregorio et al. in 2005 and Bachellier et al. 1999. *Y. pestis* and *E. coli* ERIC show a similar predicted secondary structure and since Yersinia ERIC have been shown to regulate the level of expression of neighboring genes by folding into RNA harpins is likely that this feature is conserved also into *E. coli* genome. *V. cholerae* ERIC sequences, instead, are shorter than its homologues and are predicted to be not structure. These predictions are in according with the observation made by De Gregorio et al in 2005 about the selective erosion of *V. cholerae* ERIC terminal inverted repeat that are fundamental for stem loop forming. For these reason we hypothesize that these sequence may not be directly involved in RNA

stabilization. Many families that are predicted to be structured have been found in other less studied genomes such as mycobacteria, bordetellae and pseudomonaceae. As we expect for many the predicted common secondary structure is or contains a stem-loop. In some cases the prediction is different suggesting that also structures different from the searched one has been incidentally detected. However some “noise” has to be taken into account dealing with hundred of repeated sequences. Some families feature a double hairpin (see EFA-1 and Pae-1 in Figure 18) while others have the searched stem included within a complex structure.

Genomic location of repeated sequence families

Assuming that repeats are randomly placed over the genome we can expect that since bacterial genomes is almost fully coding they fall above all within these portions. Most repeats, instead, have been reported to be located within intergenic sequences where they do not interfere with the coding information. In our study we find both families with members within genic and intergenic regions. Interestingly most families found within coding sequences (CDSs) of genomes are predicted to be not structured while most intergenic families show highly structured SLS supported also from the presence of stacked stable SLS. RANDFOLD analysis shows that 19 out of 27 intergenic families with aligned SLSs are enriched in highly structured SLSs, while this is true for only one genic family, Myp-2.

These results suggest that potentially structured families are preferentially located away from coding sequences where the translation machinery is expected to interfere with secondary structure formation while unstructured ones explain their function acting at other levels such as protein level.

Five novel families Sal-2, Myt-5, Bhal-2, Clot-2 and Clot-3 are composed of small direct repeats called CRISPR that are very abundant in bacteria and archaea. In some cases these

repeats show a dyad symmetry that can be recognized as SLS. These repeats have been recently demonstrated to play a fundamental role in bacterial resistance against viral infection by acting as a RNA interference-like system [Barrangou et al. 2007]. Also three novel intergenic structured families, Hin-1 in *H. influenzae*, Nem-4 in *N. meningitidis* and Pam-1 in *P. multocida* are composed of similar sequences, characterized by the repetition of short, abundant oligonucleotides, known as DUS [Davidsen et al. 2004]. As well as for CRISPR sequences, the repetition at short distance of DNA stretches shorter than the searched pattern produces a stem loop larger than the threshold. These sequences are required for natural genetic transformation and since they are preferentially located within or near to genome maintenance genes, they are thought to be involved in recovery of genome preserving functions. A work aimed to detect putative transcriptional terminator has evidenced that in some species terminator hairpins are indeed frequently formed by closely spaced, complementary instances of exogenous DNA uptake signal sequences [Kingsford et al. 2007].

Some novel structured families are located within coding sequences. They often contain repetitive motifs of one or a few coding regions, such as Lac-1 in *L. johnsonii*, Pae-3 in *P. aeruginosa* and Efa-2 in *E. faecalis*. The Cod-2 family, instead, even if show a very conserved sequence encodes different peptides being located in different frames. Cod-2 repeats resemble repetitive sequence elements found by Claverie and coworkers in protein coding genes of *R. conorii* [Claverie et al. 2003]. These repeats have been supposed to be involved in de novo creation of long protein segments by repeat insertion.

Five genic families found in *M. pneumoniae* are part of large (1.5-5.4 kb), possibly mobile repeated DNA sequences having coding capacity [Himmelreich et al. 1996].

About one third of the identified families are found to be “unstructured”. These sequences were not the object of the original search; a possible explanation of their detection is the incidental presence of SLSs within large repeated sequences. Most such families fall

within CDSs (see Table 4, and Myt-10 in Figure 17 as an example). Ten of them are contributed by only two genomes: *M. tuberculosis* and *M. pneumoniae*. Other unstructured families are clustered within the same CDS (Bor-3 and Bor-6 in *B. bronchiseptica*) or are dispersed within multiple CDSs, sharing a common protein domain (Bor-4 and Bor-5 in *B. bronchiseptica*, Pae-2 and Ppu-3 in *P. aeruginosa* and *P. putida*, respectively).

Genome assembly by “scaffolder”

The de novo sequencing of two relatively large bacterial genomes (5.5 and 12 Mb), was carried out in our laboratory by using a 454 GS20 sequencer and is described elsewhere (manuscript in preparation). In both cases high coverage (at least 20-fold) sequencing failed to generate a single genomic sequence with standard tools, but produced a few hundred (or thousand for the larger one) contigs. This experience prompted us to develop methods that could guide the final assembly by integrating both computational and experimental techniques, methods that have been implemented and are made available to the user through a custom developed package named ‘Scaffolder’.

The large number of contigs obtained after assembly may be due to a limitation of the experimental procedure used for sequencing, i.e. some genomic portions might be altogether excluded from sequencing. On the other hand it is also possible that simpler reasons might be involved, such as the presence of repeats, and that this be sufficient to justify the observed result in terms of contig number and distribution. Starting from this assumption, several approaches were attempted, aimed to detect relationships between contigs.

Finding links by using contig similarity and coding information

In a first approach, attention was focused on sequence boundaries. The probability that two sequences end at least with the same n -mer stretch of bases, within a population of a hundred sequences from the same genome, is very low when $n \geq 10$ ($P \ll 1E-06$). As a consequence, two sequences ending with the same stretch of bases are likely to be overlapping and therefore contiguous within the genome. Search for identical n -mers on contig ends, highlighted the presence of a number of matches of 10 or more bases much higher than expected and all the identified overlap connections were confirmed by PCR experiments. The approach was successful with the earlier version of Newbler (1.0), but

subsequently, when the same analysis was performed on contigs produced with newer versions of the assembler (1.2), no such overlaps were found anymore, as software improvements in the newer version, ended up in removal of duplicated sequences at the ends of contigs.

A similar approach was used on coding genes: BLASTX of all contig 100 bases ends against all known bacterial proteins was performed, looking for matching protein-coding regions located at the ends of different contigs. The presence of different parts of the same gene split in two or more contigs was taken as an indication of contiguity and experimentally checked. This approach only turned out to be useful for 3 links connecting 6 contigs, all confirmed by PCR; however it is clearly dependent on the available protein sequences and it is conceivable that it might be more useful when protein sequences from a more closely-related bacterial genome are available as a reference. In our case no closely-related known genome was available as about half the identified ORFs within the contigs do not share similarity with protein sequences from any other known bacterial genome.

Finding links based on initial (raw) reads

The small number of connections identified by the above described methods led to investigate new methods for detecting contig relationships. Considering the high coverage reached in sequencing (20-25X), it was taken to be very likely that almost all bases of the genome had been sequenced at least a few times, and, as a consequence, that in absence of systematic hindering factors, every base was expected to be covered by several reads. Under this scenario, it was assumed that contigs fail to be connected due to excess rather than lack of links and gaps are the result of ambiguity rather than absence of sequence information. Starting from these considerations, an attempt was done to detect, among the primary reads, the ones able to support connections between contigs. To this aim, 50 bases from each contig-end were aligned to all primary reads by using BLAST. When two

different ends align in the correct orientation to the same set of reads, a connection is defined between them. This procedure was summarized in figure 19. This procedure led to the identification of 177 connections (links) supported by at least one bridging read, involving 120 out of 130 contigs larger than 100 bps. Of these contigs, 84 have coverage compatible with being a single copy sequence in the genome, while 27 are present as double and 8 as triple copy sequences. About 85% of them have at least one connected end and 75% both of them as reported in table 8. In table 9 a summary of the identified connections is reported. As might be expected, there is a gross correlation between number of linked ends and contig coverage, i.e. contigs with coverage higher than one ('double, triple, higher' table columns) usually show more than one connection on their ends.

The quality of identified links can be estimated by looking at the number of reads supporting it. 85% of the links are confirmed by at least five reads across the ends, 65% by more than ten, as shown in figure 20. Sequences obtained by joining the ends of the connected contig-ends have been aligned to primary reads by Blastalign [Belshaw et al 2005]. As shown in one example in figure 19, reads across the artificial sequence junction are in a comparable number respect to those aligned in the inner parts of the contigs. These results verify the hypothesis that gaps are not due to sequencing limitations, but to some kind of difficulty of the assembler program in assembling such multiply linked contigs. Looking at reads across the junction in detail, it is interesting to note that when high coverage contigs are involved, they are often not 100% identical: in general different subpopulations of similar reads may be observed that together configure two or more different sequence patterns (see differently colored bases in Figure 19), as would be expected from sequence variants of repeated regions of the genome.

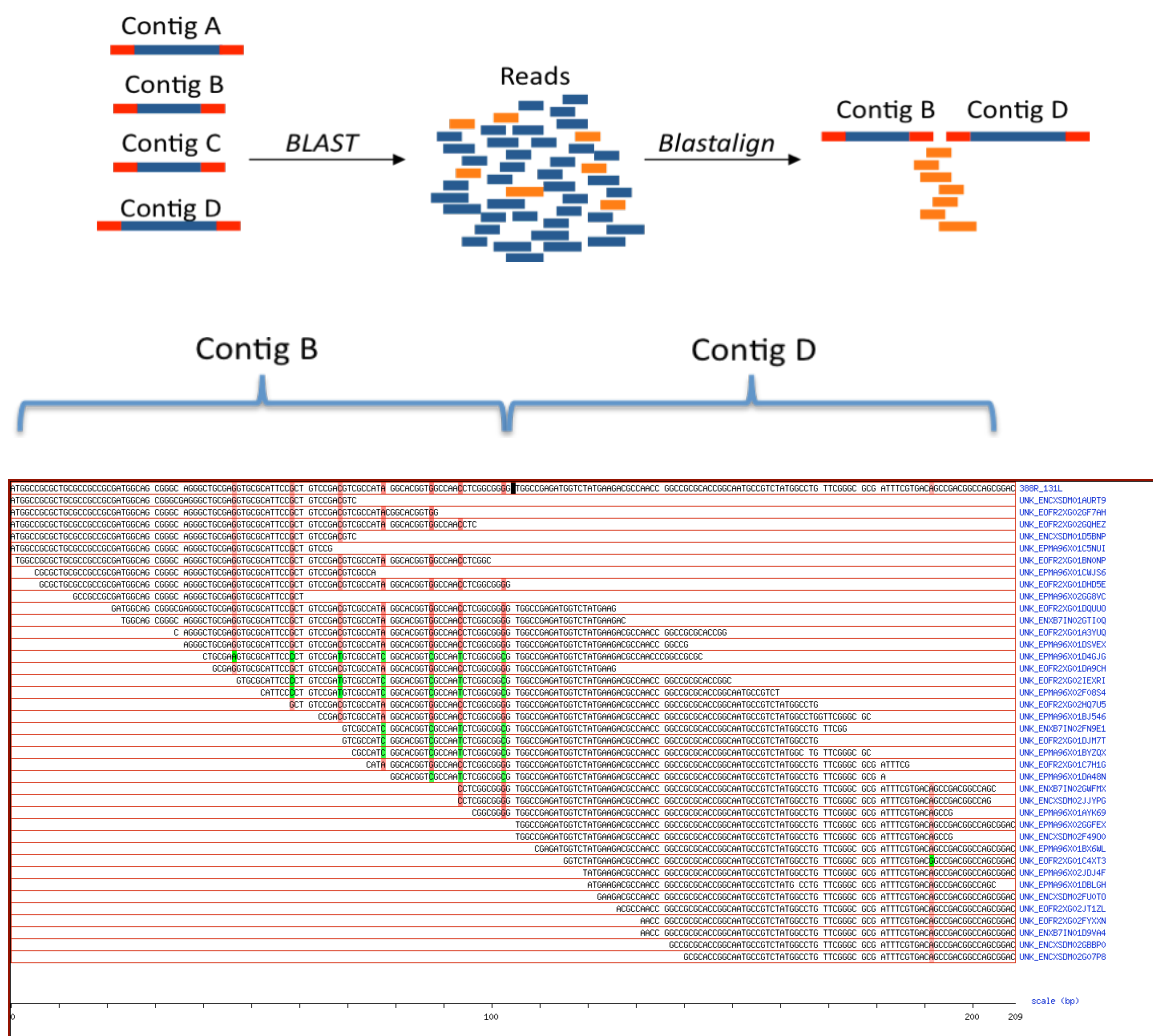


Figure 19. Finding links by BLAST

Schematic representation of procedure used to detect reads (colored in orange) across different contig ends (in red) was shown on the top. The alignment of contig ends with primary reads made by Blastalign is shown in the lower part of the figure.

n Links	Linked ends
1	132
2	70
3	18
TOTAL	220
unlinked	40

Table 8. Linked contig ends

Linked ends classified according to the number of links.

type	Coverage				total
	single	double	triple	higher	
0-0	10	0	0	0	10
0-1	15	1	0	0	16
0-2	3	1	0	0	4
1-1	52	1	0	0	53
1-2	8	0	1	0	9
2-2	4	18	5	0	27
1-3	0	1	0	0	1
3-2	0	2	1	0	3
3-3	2	3	1	1	7
	94	27	8	1	130

Table 9. Contig coverage related to link number

Coverage of contigs larger than 100 bases are reported grouped by number of links on each end ('type' column).

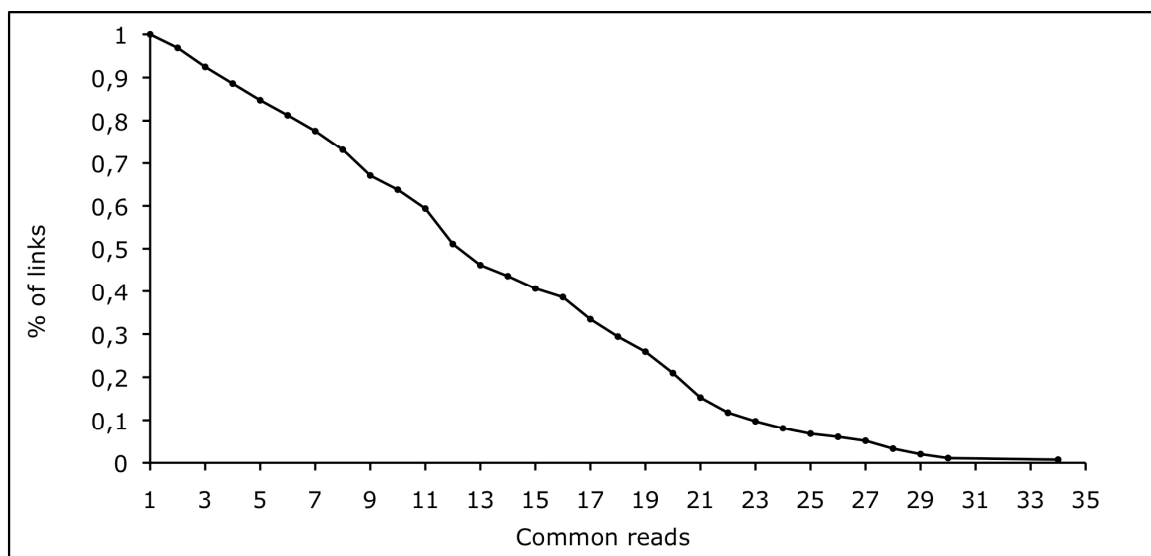


Figure 20. Link weight distribution

Link weight distribution is reported as the fraction of links supported by each number of reads (at least).

Displaying relations as a connected graph

The hundreds of contigs and links may be visualized as a graph. To this aim the Graphviz tool has been used to build a graphical representation of the contigs and their relationships between contigs (see Figure 21). Within the graph, contigs are represented as nodes and links edges. Each contig is represented as a box, whose sides are the extremities of the sequence. Ends are connected by edges, which indicate a putative contiguity on genome. On each edge the weight, i.e. the number of reads supporting this relation, is reported. Contig boxes contain information as contig identifier, length in bases, sequence coverage, both raw and normalized to the overall average coverage, i.e. overrepresentation in the genome.

Contig coverage reported in graph is estimated in the following way:

for contigs larger than the average read length L coverage is the product of L and the number of contained reads n , divided by contig length l .

$$Cov = \frac{n * L}{l}$$

for contigs shorter than L , n is used as the coverage.

A color code has been used to classify contigs according to the degree of agreement between coverage and number of connections. Classification distinguishes the following groups:

- contigs with no links;
- single coverage contigs with one link on one end;
- single coverage contigs with one link per end;
- multiple coverage contigs with the corresponding number of links on both ends;
- contigs with less links than those expected by coverage;
- contigs with links exceeding those expected by coverage.

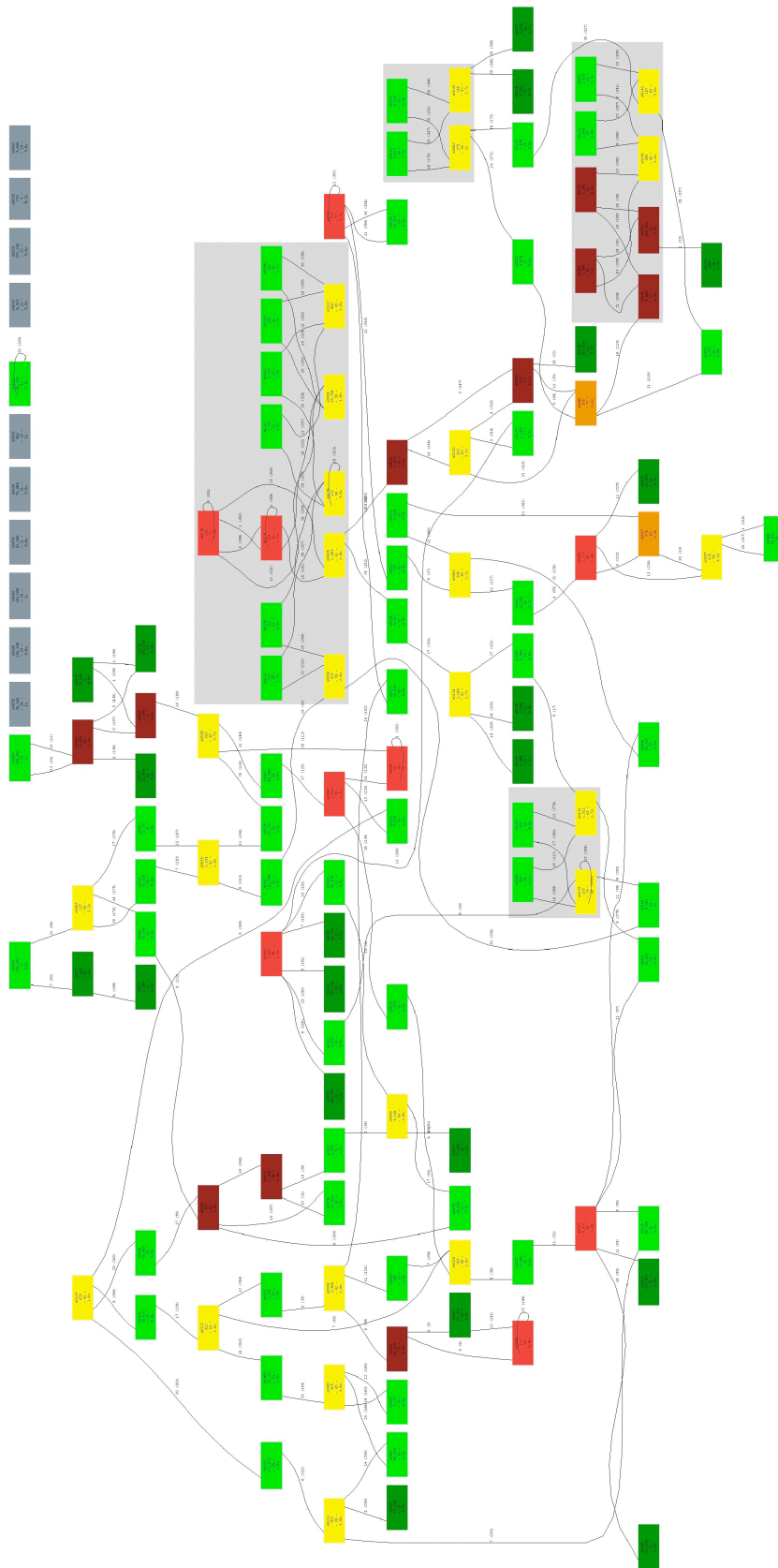


Figure 21. Genomic assembly of a 5.5 Mb bacterium as a connected graph

Contigs and their relations are displayed as nodes and edges of a connected graph. Contig color is chosen according to correlation between coverage and links as explained in Methods. In this graph only contigs larger than 100 bps are shown.

Graph analysis

Apart from a small number of isolated contigs, the majority of contigs is part of a single complex network. Contigs with a single connection per end are never connected with each other, but almost always connected with short hyper-linked, high coverage, contigs. They clearly represent repeated sequences that the assembler is unable to untangle and that are causing interruptions in long stretches of unique sequences.

One very large contig (about 49 Kb) is separated from the network and features double coverage and a single link connecting its ends in a circular fashion, as expected from a circular extrachromosomal DNA element. PCR and other experimental evidence (not shown) confirmed that the DNA molecule is indeed a circular plasmid, for which the higher coverage would predict a 2:1 stoichiometric ratio with the chromosome. All putative ORFs have been detected and translated and predicted proteins have been used to search the KEGG database for matches with known pathways by using the KEGG Automatic Annotation Server KAAS. This analysis revealed that the entire type IV secretion system, a structure homologous to conjugation machinery involved in mobilization of both plasmids and proteins was present.

Resolutions of ambiguities

The ambiguities present in the connected graph prevent the identification of a univocal path representing the whole genome sequence. In an attempt to solve them, two approaches have been tried: one based on computational analysis of primary reads and the other on PCR experiments.

Computational multiple contig separation

The small size of primary read sequences limits the possibility of using the read itself as a mean to untangle the network only to the theoretical case of contigs smaller than 100 bases. None of them was found in the course of manual analysis of a small number of

nodes, but in two cases, by following the sequence through a few reads across contig borders it was possible to univocally assign the contigs flanking a repeated contig. This observation was used as the base for the development of a computational tool able to extend this approach to larger multiple contigs, where the aligned reads contain an uninterrupted path of micro-heterogeneity as the one described in figure 22.

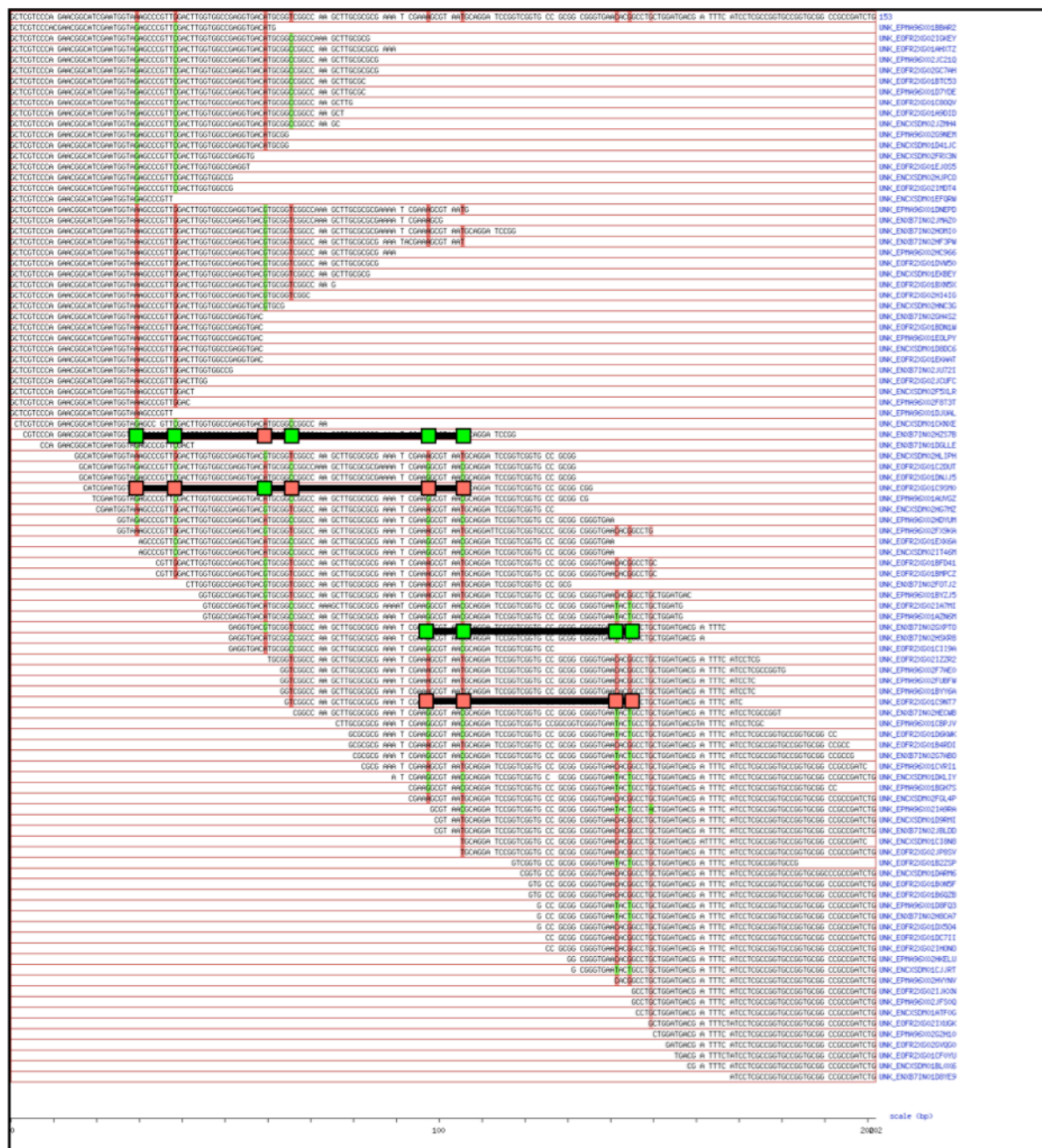


Figure 22. Alignment of a high coverage contig with primary reads

The alignment of a high coverage contig with primary reads detected by BLAST is shown. Micro-heterogeneities are highlighted by coloring the bases in red and green.

A scanning algorithm has been used to develop a software tool able to solve an alignment of reads mapping within a multiple contig and generate the sequence components by taking advantage of micro-heterogeneities, i.e. column in which two or more different nucleotides are consistently present in primary reads. The software procedure was run on several multiple contigs, and resulted in separation of the contig into its sequence components. An example is reported in figure 23.

The algorithm is designed to separate the two or more sequence variants combined into a contig. In the simplest situation, i.e. when no micro-heterogeneities are found, only one variant is reported. Alternatively, when a sequence heterogeneity is found at a given position, the procedure creates a number of sequence variants equal to the number of bases observed in that position and assigns each overlapping read to the respective variant. For example if within the n th-column of alignment G and T bases are alternatively present in different reads, the procedure creates the variants N and $N+1$ respectively containing the G or T bases. A threshold indicating the minimum number of reads supporting the evidence of a new variant is a parameter configurable by the user. During alignment scanning, the reads previously assigned to a variant are used to guess the base to be expected in the new alignment position. In most cases the observed bases agree with the expected assignment, i.e. in a given position the same base is found in all reads assigned to the same sequence variant. Only when a new micro-heterogeneity is found the procedure creates new variants and assigns reads to them. When a new read is encountered during the scan process, it remains unassigned as long as no micro-heterogeneity exists, but as soon as one is found, the read is assigned to a specific variant identified according to the heterogenous base. In this way, the method is able to follow sequence variants along the alignment. However, it is possible that all reads assigned to a variant end before a new heterogeneity is found, thus making impossible any further extension of the variant. In this case, the procedure stops

the separation of current variants and creates a new multiple contig representing the region, linked to the previously detected variants.

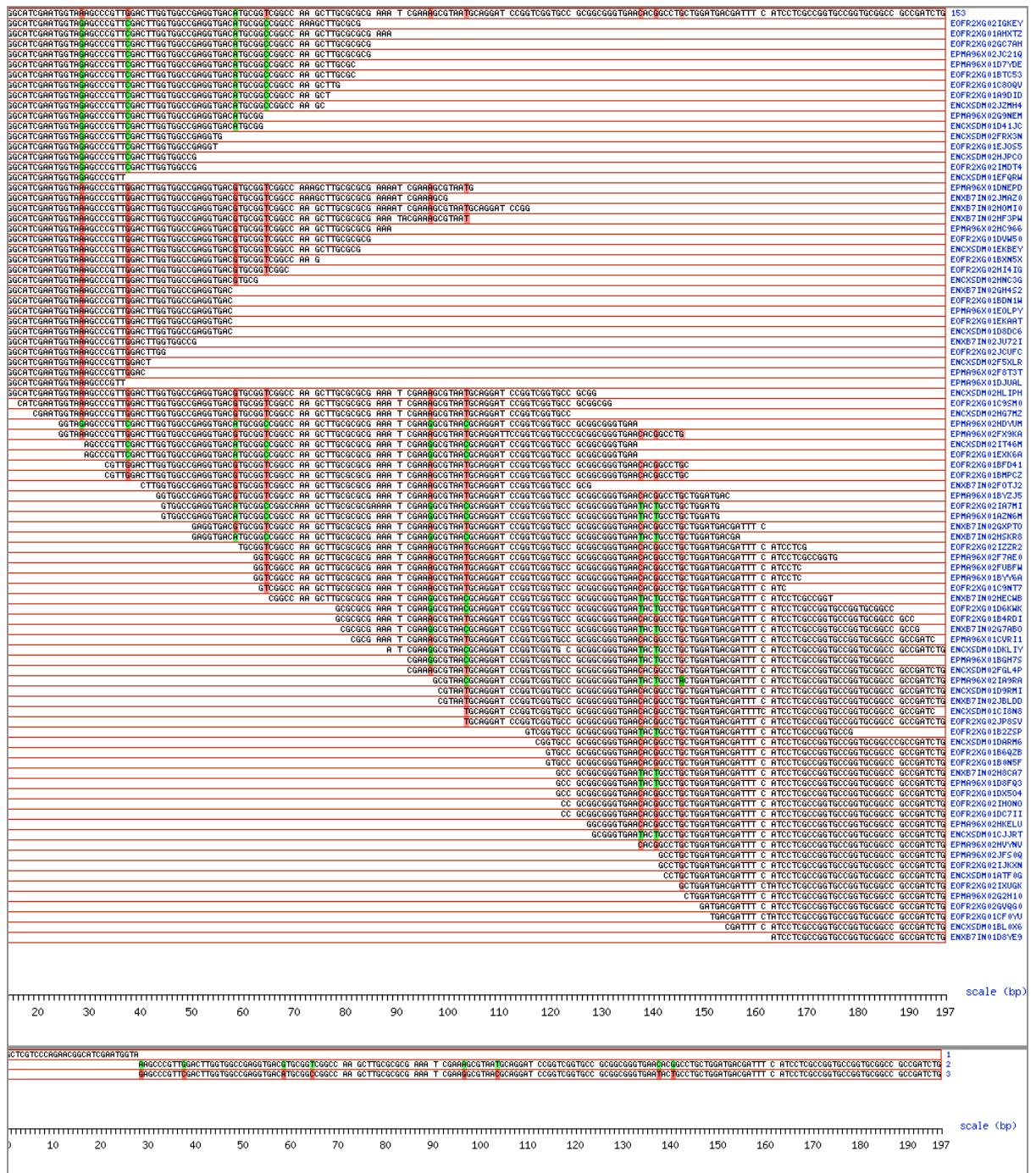


Figure 23. Solving a repeated contig by micro-heterogeneity analysis

A high coverage contig sequence is aligned to primary reads. Bases not conserved along the alignment, i.e. micro-heterogeneities, are differently colored depending on their abundance, in the following order from most to least: red, green and blue. The presence of two different sequence patterns indicated by alternating colors is highlighted.

This multiple region in turn ends when a new micro-heterogeneity is found, and new variants linked to it are generated. In this way, a multiple contig is completely separated into its components, or, as often observed with larger repeats, it is divided into a variable number of multiple contigs, linked by two or more dissimilar sequences. Application of this procedure was able to successfully solve a substantial number of ambiguous contigs. Even when one long multiple contig could not be completely separated, it was still possible to reduce it into fragments of smaller, more manageable size.

Resolution of ambiguities by experimental methods

No matter how well-behaved the computational approach is, there are situations where experimental methods are required, often in the form of PCR amplifications. Typically PCR experiments are used:

- to validate the connection of two contigs predicted to be neighbors within the genome;
- to untangle situations in which many contigs are linked to both ends of a multiple one;
- to identify neighbouring contigs by combinatorial PCR within a limited set of non-connected contigs

The experimental approach based on combinatorial PCRs is always applicable, in principle, but it easily requires an exceedingly large number of reactions: considering only the 92 contigs longer than 1000 bases produced by the assembly of the smaller genome this approach would require 184 primers and 4186 PCR reactions. In practice this number was reduced by only verifying contig connections identified by one of the procedures described above, and using the combinatorial approach only on the few remaining unconnected contigs. Sequencing of the amplicates by Sanger method was used to validate the experiment and correct errors in contig end sequences.

To this aim a strategy to assist in the design of PCR primers was developed. Primers are searched in contig-ends that do not match on other involved contigs by using the eprimer3 program from Emboss package and then PCR experiments are simulated with calculated primers on each contig combination by using the PrimerSearch program (always from Emboss package). The procedure was designed to also give additional information useful to experimental design like the primer GC content and length, the melting temperature and the expected product length. An example of PCR experiments design is reported in figure 24. Leaving some parameters such as primer GC content, melting temperature, length and ability to prime on other contigs freely modifiable allowed to design a great number of primers and to fill a great number of gaps. In some cases it was not possible to find a good primer that univocally recognize one contig because flanking contigs have very similar ends. To solve also these situations an extension of previous procedure was implemented. The new procedure detects identical regions between contigs flanking one side of the multiple one and produces a single common primer. In this way the problem is solved by analyzing small variations in the amplificate sequences. The application of this strategy is not restricted to the presence of near identical contig-ends and allows reducing experiments to the number of contigs with different primers. In order to distinguish these two procedures the former was named X model and the latter Y model and are schematically explained in figure 25.

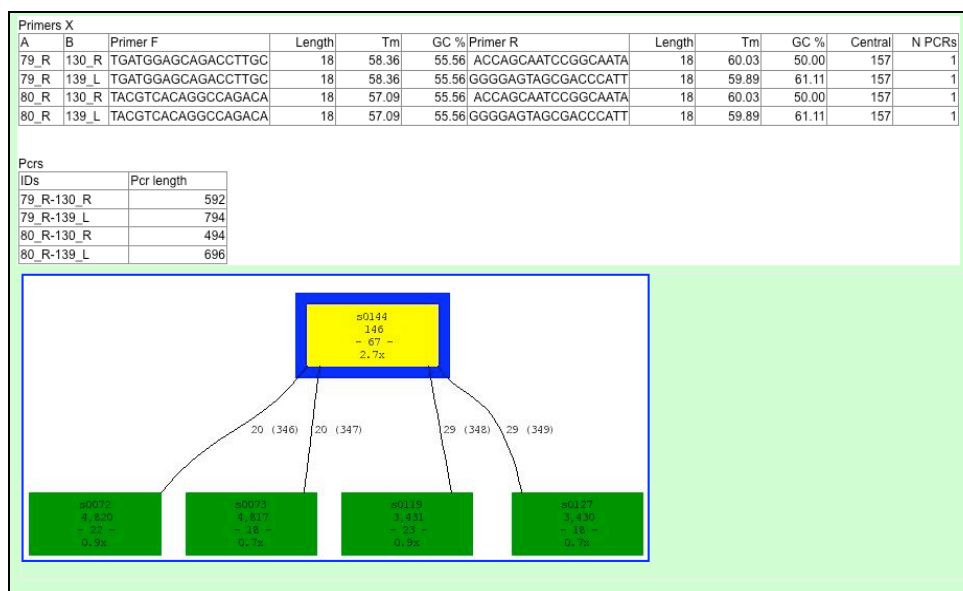


Figure 24. Design of PCR experiment

The figure reports the primers (upper) calculated to untangle the contig network reported in the graph (lower). Sequence, length, melting temperature, GC percent of both forward and reverse primers together with number of predicted PCR (“N PCR”) are shown, together with length of PCR products predicted for each contig combination.

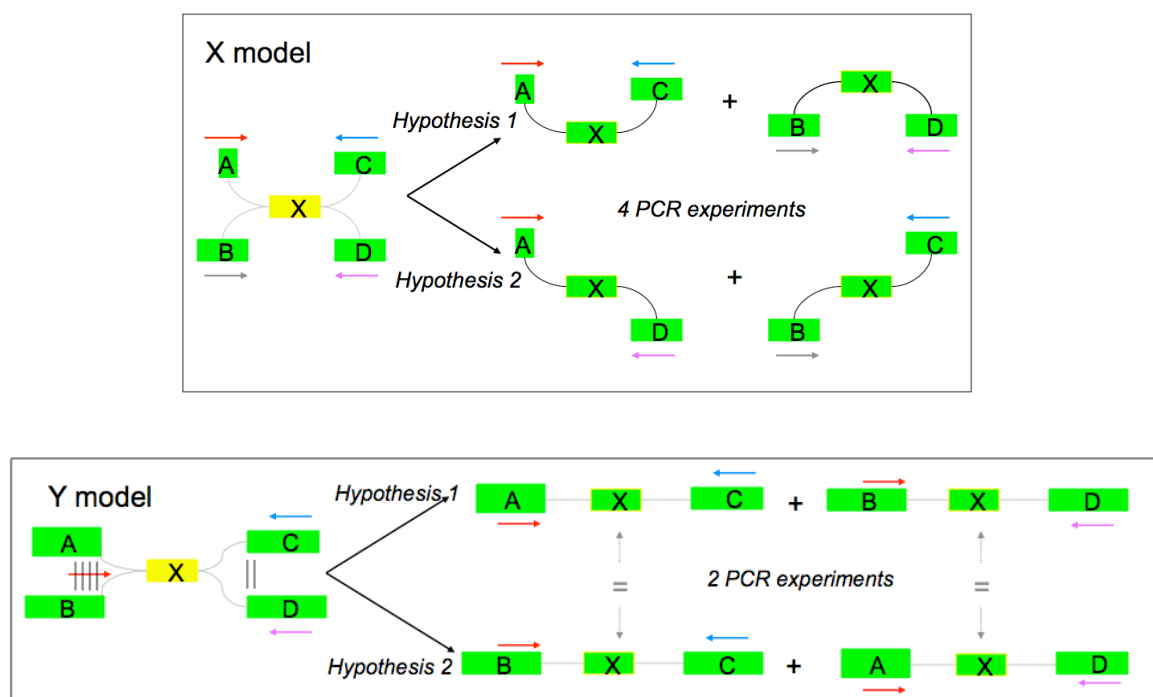


Figure 25. Solving ambiguities by using PCR experiments

PCR experiments have been used to untangle two contig networks. Green and yellow boxes indicate single and double covered contigs; colored arrows represent the primers required for each model.

Scaffolder tool

All the procedures described above have been implemented into software tools combined into a package named “Scaffolder”. The package is designed to assist the researcher in *de novo* sequencing projects, by starting from a set of unconnected contigs and is able to detect links between contigs and solve most ambiguities deriving from repeated sequences. Scaffolder guides the overall assembly process by linking contigs into a multi-connected net, separating repeated sequences by a computational approach based on sequence micro-heterogeneities and selecting primer pairs to experimentally verify predicted links and untangle zones that cannot be computationally solved. It uses several different tools for performing the analyses, such as BLAST, and relies on a relational database management system (RDBMS) for storing both the initial data and the subsequent results. It also integrates an automatic versioning system of the assembly that allows following the quality and assessment of the sequences during the assembling procedure over time. Scaffolder is organized into independent modules, aimed to:

- access the DB-stored data (storage engine);
- analyze and edit the assembly (computing engine);
- manage subversions;
- communicate with the user through both a command-line and a web interface.

The system is written according to the object oriented programming paradigm and is mostly implemented in PHP. The code is written as a number of objects, mostly specific of the various modules implemented, except for those that integrate command line tools and that act as database interface.

Storage engine

The storage engine is a module that communicates with the database for accessing or uploading data regarding primary reads, scaffolds, links and assemblies. It is designed as a

single object whose methods can be accessed only by the computing engine. It does not directly access the relational database, but uses independent objects specifically designed for communicating with generic RDBMS. The storage engine is able to manage and store DNA sequences with associated quality and lengths as when handling reads or scaffolds, scaffold ends involved with the relative weight when handling with links. It also automatically calculates the coverage and average read length for a specific assembly. Moreover, the storage engine keeps track of the assembly subversion and stores every operation that modifies the assembly into logs (see below).

Scaffold analysis

Scaffolder implements the previously described methods as procedures aimed to analyze contigs, scaffolds and links and to guide the design of experiments for validation of the predicted relations. The computing engine is composed of one object embedding all the procedures and is connected with the storage engine for retrieving and uploading data. Some of the procedures call external tools, as BLAST or PrimerSearch.

The computational engine of Scaffolder allows identifying links between contigs (see “Finding links based on initial (raw) reads” paragraph) and draws the graph of scaffold relations (described in “Displaying relations as a connected graph” paragraph). The graph is implemented as a clickable map where any object may be selected with its neighbors to create sub-graphs that include all the objects directly connected to it up to a given “depth” (see Figure 26).

Visual inspection of large graphs can give an idea of global connectivity, but it is not suitable for statistic purposes. For this reason, the computational engine of Scaffolder implements methods for displaying scaffolds also in a tabular way together with information, such as length, coverage, number of connections and number of reads. An example is reported in figure 27.

In order to solve ambiguities, make computational analysis or drive experimental design, several functionalities have been implemented, such as those that allow identifying and automatically aligning initial reads to one scaffold sequence or to two contiguous scaffold ends. The alignments are displayed in color, to emphasize the presence of micro-heterogeneities (see for example Figures 19 and 22), and can be fed as input to the micro-heterogeneity analysis tool, to separate sequence variants starting from a multiple coverage contig.

PCR experiments can be used to validate putative links or untangle the connections of contigs flanking a repeated one. The automatic procedure of designing and testing the primers by both X and Y models (described in “Resolution of ambiguities by experimental methods” paragraph), is implemented as operations (functions) that allow designing primers between two linked scaffolds or on each end of a single scaffold.

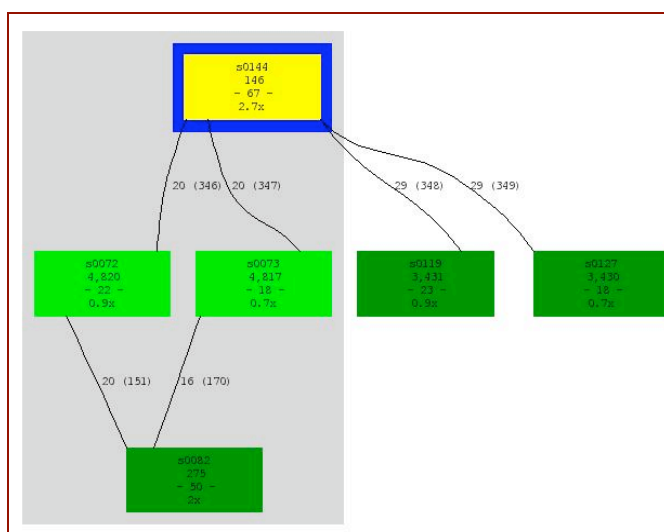


Figure 26. Assembly of a restricted number of contigs as a subgraph

A subgraph indicating contigs and their relations is shown. The graph is built starting from the contig highlighted by a blue box and is extended to the contigs connected to it following a depth index. This subgraph was built by using a depth index of 2.

Num	ID	Coverage	Length	Reads	links	
1)	157	67	146	92	2-2	<input type="button" value="set"/>
2)	91	50	275	128	2-0	<input type="button" value="set"/>
3)	130	23	3,431	734	0-1	<input type="button" value="set"/>
4)	79	22	4,820	976	1-1	<input type="button" value="set"/>
5)	80	18	4,817	803	1-1	<input type="button" value="set"/>
6)	139	18	3,430	576	1-0	<input type="button" value="set"/>

Figure 27. Displaying contigs in tabular way

The contigs in figure 26 are reported in a tabular way together with their coverage, size in bases and number of reads used for assembly. The number of links on each contig end is reported in column “links”.

Editing the contigs

The operations for editing the assembly are structured as a three level hierarchy. With respect to the complexity of the action they have to do, they are classified as low-, middle- and high-level operations, where higher operations work by using the lower level ones.

The low-level operations consist of functions that perform basic and simple actions, such as creation or deletion of links and scaffolds. They can be called directly from the user in order to execute simple tasks or from the higher-level operations as part of more complex instructions.

Middle-level operations consist of tasks involving one or two objects. Essentially they are referred to as joining flanking scaffolds and splitting a repeated scaffold into more copies. The joining process consists of three consecutive low-level steps: two deletions of linked scaffolds, followed by the creation of a larger scaffold, whose sequence is obtained by combining those of the deleted ones. The splitting process, instead, consists of the creation of a copy of the involved scaffold, optionally having in tow the creation of a couple of links from the parent.

Finally, the high-level operations allow the execution of complex tasks that consists of a combination of middle- and low-level ones. For example, the “Split-and-Join” operation

splits a repeated scaffold and join its copy to a couple of flanking scaffolds by using specific links which are in turn deleted at the end of the process. Another example is turning a scaffold into objects connected by links, as a result of micro-heterogeneity analysis.

Version management

The execution of each assembly editing operation changes the assembly in terms of number of scaffolds and links, producing a sort of evolution history of the assembly itself. In order to keep track of the overall scaffolding procedure, an automatic versioning system has been implemented, where each operation ends with the definition of what is called a new “assembly subversion”. As a consequence, contigs, scaffolds and links are originated within a subversion and killed in another one. The initial set of contigs are assumed to be born at subversion 1. Every scaffold is “alive” until it is fused with others or discarded for other reasons. In this way at the end of the assembly procedure, the number of scaffolds still alive corresponds to the genomic elements composing the genome, for example chromosomes. Scaffolder can show and analyze every assembly stage, only visualizing the elements of a particular subversion. In this way, the assembly process may be followed over both time and operation by creating graphs for each subversion.

Scaffold history

Each scaffold, obtained by using one or more hierarchical processes, is the result of a variable number of steps of splitting and fusing “parental” contigs or scaffolds. By using the subversioning system, it is possible to follow the story of every single scaffold, in terms of which are its ancestors. In order to do this and to describe all the steps that conduce to the formation of a given scaffold, a specific method has been implemented. Given a scaffold, it produces a historical graph over time (subversions) where nodes are the relatives, i.e. from most ancient to the selected one, and edges the parental relations.

The graph is built based on the data that the storage engine automatically stores in a dedicated table of database when scaffolds are created by fusion or duplication of others (see Figure 28). Moreover it is possible for a given a scaffold to create a list of all contigs used for its assembling. When the assembling process is able to produce the final sequence this list indicates how the initial contigs are located in the genome.

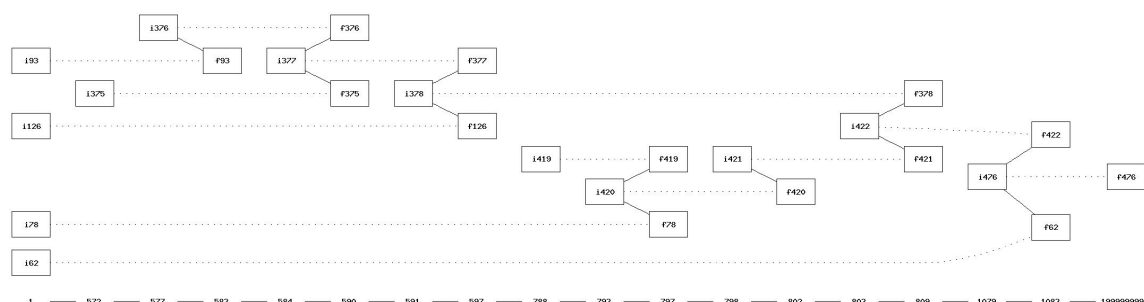


Figure 28. Scaffold history

All the operations that are involved in building of scaffold 476 starting from initial contigs 93, 126, 78, 62 are reported as described in Results. Each box represents a contig that is flagged as “i” if is created or “f” if is deleted during subversion indicated below.

Assembly progress

As all operations are logged, an estimation of the duration of the whole process can be easily obtained. The trend of the scaffold merging process over time depends on many factors, such as the number of gaps to be filled, the number of identified links and linked-ends, the number of linked elements and isolated ones, the number of high coverage objects with multiple connections at their ends, which need to be tested. All these factors have been combined into a “scaffolding score”, which is automatically calculated at the end of every subversion-step. This index is the sum of two scores, related to scaffolds and links respectively. The “scaffold score” gives an estimate of the number of scaffolds by taking into account scaffold size and coverage:

$$\text{Scaffold score} = (0.01 * (\text{short} + \text{lc}) + \text{sc} + 1.5 * \text{dc} + 2.5 * \text{hc})$$

where **short** is the number of scaffolds shorter than 100 bases, **lc** is the number of very low coverage, probably erroneous, scaffolds while **sc**, **dc** and **hc** is the number of scaffolds with single, double or higher coverage respectively.

The “link score” indicates the assembly connectivity rate is high if few connections are detected and goes down to 0 when the assembly is fully connected:

$$Link_score = \left(\frac{2(links)}{le} - 1 \right) * (scaffolds)$$

where **links** and **scaffolds** are the total number of links and scaffolds, while **le** is the number of linked ends

In figure 29 is reported the scaffolding score variation of our genome project.

Date	Subversion	10-100	101-1000				1000	Tot	Plasmids	Links	Linked ends	Score
			Low	Single	Double	Triple						
April 23 2008	1	97	16	37	24	9	92	242	1	177	248	291.1
April 30 2008	721	97	16	14	10	5	65	192	1	105	148	187.1
May 6 2008	787	97	16	13	9	5	60	186	1	97	136	178.5
May 9 2008	827	97	16	12	8	4	58	183	1	93	130	171
May 12 2008	876	97	16	12	8	4	55	180	1	89	124	167.5
June 9 2008	958	97	16	11	7	4	50	174	1	81	112	159.3
June 10 2008	1075	97	16	10	6	4	44	167	1	68	97	140.3
June 11 2008	1181	97	16	8	6	4	41	162	1	59	87	125.9
June 13 2008	1204	97	16	8	6	4	40	161	1	108	107	231.1
August 6 2008	1281	94	16	7	6	4	38	155	1	98	95	228.9
August 7 2008	1312	93	16	7	6	4	36	152	1	89	89	214.1
October 1 2008	1341	93	16	7	6	4	34	150	1	84	85	206.6
October 2 2008	1382	92	16	6	5	4	32	146	2	77	77	200.6
October 6 2008	1489	91	16	4	3	3	25	136	2	55	61	149.3
October 7 2008	1511	91	16	4	3	3	24	135	2	54	60	147.1
December 12 2008	1611	89	16	3	3	3	18	126	2	39	42	140.1
February 16 2009	1726	88	16	2	2	2	15	121	2	21	30	72.4
February 18 2009	1759	88	16	1	1	1	14	119	2	17	26	54.7

Figure 29. Progression of the assembly of a 5.5 Mb bacterium in time

Each step in which assembly was modified is reported together with the subversion, the number of scaffolds grouped in classes according to length and coverage. Number of links, linked ends and score as explained under Results are reported.

Interfaces

The scaffolder package consists of a set of objects that may be accessed in two ways: as a command-line tool or via web interface.

The command-line tool is composed of a wrapper that calls the computational engine and gives full access to the implemented methods. It is the best way to integrate Scaffolder in a more complex pipeline. With single commands it is possible to use it for retrieving the contigs as map or table, to align reads to a scaffold, to calculate PCR primers, or to perform operations on scaffolds. It may also be used to re-run the assembly process in an automatic way. The full specifications are given in Table 11.

```
Available options:
[-h --help]          This Help
[-v --version]       Display version
[-c --cmd]           string
                    Command to be executed. Allowed values:
                    help
                    doScaffList
                    doXprimers
                    doYprimers
                    doAlign
                    doSimpleVersion
                    doCompleteVersion
                    calcVersions
                    contigs2scaffold
                    blastOnScaffolds
                    deleteScaffold
                    deleteLink
                    splitScaffold
                    splitScaffoldAndJoin
                    unbundleScaffold
                    joinScaffolds
                    createLink
                    stopLinks
                    explodeScaffolds
                    createNewLinks
                    setPlasmid
-i --iniFile         string
                    Project file name
                    REQUIRED
[-q --inFile]        string
                    Input query file

[-o --outFile]       string Write output into file
[-s --scaffold]      list A comma separated list of ScaffoldIDs to
perform command.
[-l --link]          list A comma separated list of LinkIDs to perform
command.
[-e --ends]          list A comma separated list of ends to perform
command.
```

<code>[-a --assembly]</code>	integer Assembly. Default = (the most recent one)
<code>[-u --subversion]</code>	integer Subversion. Default = (the most recent one)
<code>[-d --depth]</code>	integer Link depth. Default = 1
<code>[-p --prefix]</code>	string Prefix for outfiles command. Default = tmp/res
<code>[-m --length]</code>	integer Minimum scaffold length. Default = 100
<code>[-r --coverage]</code>	integer Minimum scaffold coverage. Default = 4
<code>[-P --prMinLength]</code>	integer Primer minimum length. Default = 18 Allowed values:{10-20}
<code>[-g --prGcPercent]</code>	integer Primer GC%. Default = 50 Allowed values:{30-70}
<code>[-x --prMaxPriming]</code>	integer Primer max priming. Default = 12 Allowed values:{5-100}
<code>[-b --prBorderSize]</code>	integer Scaffold border size. Default = 500 Allowed values:{50-2500}
<code>[-E --evaluate]</code>	integer Max BLAST evalue. Default = 0.01

Table 10. Options available by using the Scaffolders command line

Web interface

The web interface grants access to Scaffolders methods in a user-friendly and intuitive way. The webpage uses a control bar to give access to all analysis tools and includes an area to display results. The control bar is composed of three main panels. The “Data Set” panel is used to manage different genome projects and assemblies. A subset of scaffolds may be chosen according to length, coverage and weight of connectivity. This panel also allows moving through the various subsversions of the assembly process. The panel “Mode” is used to select and display scaffold subsets. The third panel gives access to functions that modify the assembly state such as deleting subsversions, creating and deleting BLAST-based links and to execute BLAST analysis on a given subset. When a view mode is selected, an additional panel appears containing all the relevant parameters, for example when the graph mode is selected the “Map” panel appears with controls for changing graph dimension or scaffold display mode. The web page is shown in figures 30 and 31.

The graphical view is active and within it scaffolds can be selected by clicking on them. In this way the analyzed subset is reduced to the selected scaffold and its close neighbors up to a given depth index. Scaffold subsets may also be displayed as a table. Once a scaffold is selected, a new panel indicating the available operations appears, where all the previously described operations such as analysis of micro-heterogeneity or alignment of primary reads to scaffold sequence are easily accessible. This panel also gives access to PCR design and result evaluation (Figures 24 and 32).

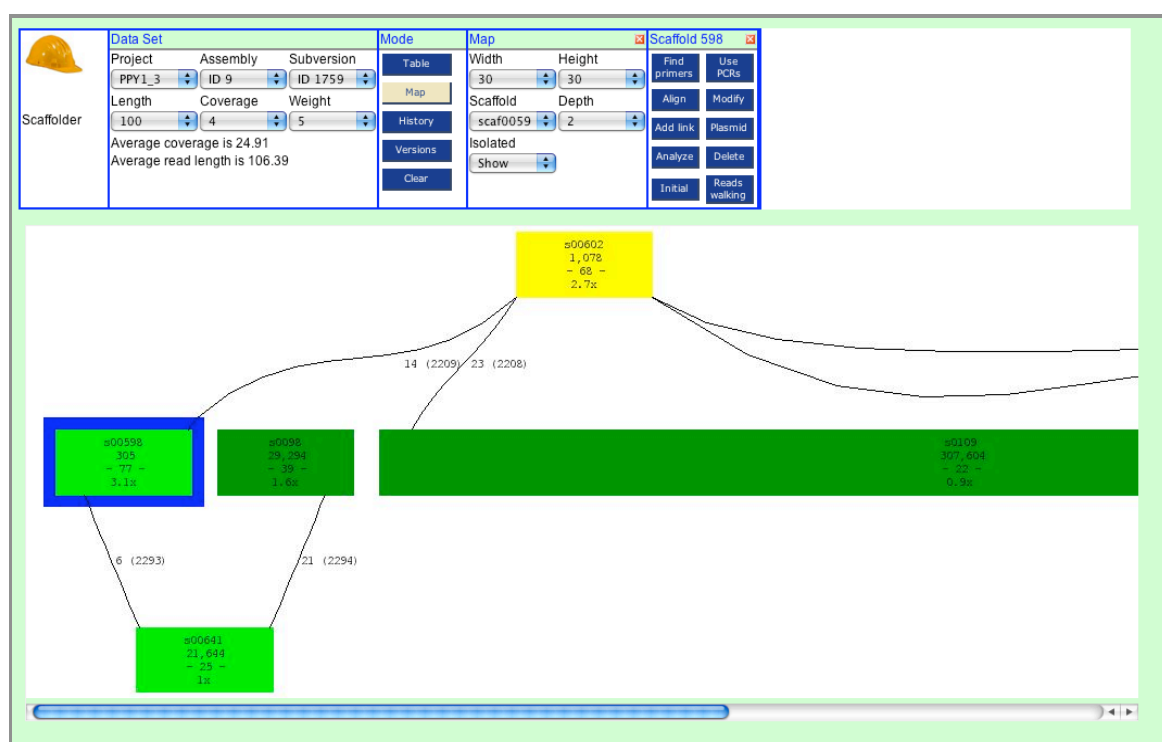


Figure 30. Web interface for Scaffolder (1)

The web interface for Scaffolder is shown. A control bar at the top hosts boxes for groups of related controls. In each box buttons and menus are used to access the various program functions.

The screenshot displays the Scaffolder web interface. At the top, a 'Data Set' panel shows 'Project: PPY1_3', 'Assembly: ID 9', and 'Subversion: 1'. Below this, 'Scaffolder' statistics are shown: 'Average coverage is 24.91' and 'Average read length is 106.39'. A 'Mode' panel on the right includes buttons for 'Table', 'Map', 'History', 'Versions', 'Clear', 'Find primers', 'Align', 'Add link', 'Analyze', 'Initial', 'Use PCR', 'Modify', 'Plasmid', 'Delete', and 'Reads walking'. The 'Align scaffold to reads' window is open, showing 'Min aln length: 20' and 'evalue: 0.01', with 'alignment type' set to 'megablastalign'. Below these panels, the 'Sequences Blast result Alignment' table is visible, showing a list of sequences aligned to Scaffold 157. The table has two columns: 'Sequence' and 'Alignment'. The 'Sequence' column lists various scaffolds (e.g., TCACCCGGA, TCACCCGGA, TCACCCGGA) and the 'Alignment' column shows the corresponding alignment (e.g., 157, 157, 157).

Figure 31. Web interface for Scaffolder (2)

The control bar in “Scaffolder” mode: once a scaffold is selected it gives access to all the available operations. Selection of “Align” button in this module produces the reported alignment of the scaffold sequence with its primary reads.

Methods

Selection of highly repeated SLS

Initial SLS population is represented by sequences detected in Petrillo et al. 2007 conducted on 40 bacterial genomes. A subset of sequences predicted to fold into a stem loop structure (SLS) with a free energy ≤ -5 Kcal/mol was analyzed for this study. Clusters are obtained by using BLAST [Altschul et al. 1990] and MCL programs [Enrigh et al. 2002]. An all-against-all BLAST comparison was performed on the SLS population within each genome to create E-value based distance matrices. The resulting matrices were pruned by removing links caused by overlapping SLSs and subsequently fed to MCL program that produces a set of clusters. BLAST was performed with an E-value cut-off of $1E-4$ and forcing only search on the top strand sequence. The MCL inflation parameter (I) was set equal to 4 to have a stringent condition. The alignments of clustered elements were produced by PCMA [Pei et al. 2003] by using default parameters. ALISTAT was used to analyze alignment within each cluster by using default parameters.

Analyzing stability of SLS predicted secondary structure

The probability of original and repeated SLSs and control sequences to form a stable secondary structure was tested by running RANDFOLD tool [Bonnet et al. 2004]. The shuffling used to create the random distribution was performed by preserving the dinucleotide frequencies by using the ‘-d’ option. RANDFOLD was set to compute 1,000 randomizations for each sequence. In the tests reported in figure 1, all clustered SLSs (panel A) were compared to a original SLSs represented by the 5% of initial population (panel B) and to a number of genomic sequences having the same size of clustered SLSs, randomly extracted from the corresponding genomes (panel C). Control sequences analyzed in panels B and C, were selected three times, in order to evaluate average and standard deviations.

Regrouping clusters in larger families

The regrouping procedures summarized in Table 2 were made as follows:

- 1) Regrouping by sequence was made by using the BLAST-MCL procedure described previously on all SCRs, but in a less stringent way. An inflation parameter of 1.4 was used.
- 2) Regrouping by strand was performed by using again the BLAST-MCL procedure, but allowing this time searches on the complementary strand. The inflation parameter was set to 1.4.
- 3) Regrouping by location was obtained by joining clusters with SCRs partially overlapping or flanking, according to their genomic coordinates. The maximum distance allowed in flanking definition was of 150 bp.

For each regrouping procedure was defined a group of clusters when it contains at least 50% of the elements derived from original clusters.

Extension of families members by cycles of HMM searches

Extension in size and number of detected families were performed by using a procedure based on cycles of alignment by PCMA and search on the genome by HMMER package tools [Bateman et al. 1999]. In first iteration SCRs of clusters regrouped by sequence (see Table 2) were aligned by PCMA with option 'ave_grp_id' set to 50 and then alignment were fed to the procedure described as Fig

follows:

- 1) Each alignment is used to build a HMM by HMMBUILD and then it is calibrated by using HMMCALIBRATE with the default options.
- 2) The produced HMM is used to search sequences on the genome by using HMMSEARCH with an E-value cut-off set to 1E-10. Independent searches are run on each genomic sequence strand.

- 3) Identified sequences are extracted and aligned to their parental HMM by HMMALIGN. When overlapping sequences were selected on opposite strands the one with the worse score and E-value was discarded to avoid repeated search.
- 4) The aligned sequences are extended by attaching to them the neighboring sequences on the genome up to 10% of the parental HMM size and are aligned by using PCMA.
- 5) A new model is build starting from the alignment of the extended sequences and is fed again to the procedure returning to step 1.

The iterative procedure ends when one of the following criteria is met:

- The detected sequences that cover the entire model are less than 7;
- The extended alignment is not able to produce a new HMM, larger than the previous one (within a tolerance of 3 bp).
- The alignment contains a number of gaps higher than 30% of the aligned bases.
- The extreme value distribution, derived from the model calibration, is in the range $\text{Average_Score} \pm 3 * \text{Standard_Deviation}$, derived from HMMBUILD.

When the procedure ends the obtained HMM and the final alignment are used to define the family characteristics.

Secondary structure analyses

All SLSs contained in sequences of each family were tested by RANDFOLD as described previously and considered as stable if their p-value is < 0.005 . Families were classified according to the fraction of sequences containing at least one positive SLS. Four categories indicated as, '+++', '++', '+' and '-' indicate respectively a fraction of stable SLS of 90% or above, 70-90%, 50-70% and less than 50%. Representative sequences of the families were chosen to perform other structural analyses in the following way:

- 1) All sequences able to match the entire model are sorted by the E-value determined by HMMSEARCH.

2) Six sequences are picked from this population by selecting the best model-fitting one and five more, if available, with progressively increasing of the E-value.

Sequences were aligned to parental HMM by using HMMALIGN and the resulting alignments were analyzed by RNAz (version 0.1.1) [Washietl et al. 2005].

Alignments with length ≤ 200 bp were used as a single block in RNAz analysis, while alignments longer than 200 bp were screened in sliding windows of length 120 and 40 slide, according to the procedure described by Washietl et al. 2007.

RNAz was used with default parameters. All alignments with classification score $P > 0.5$ were considered as positive. Hits from overlapping windows were analyzed again by using larger sliding windows.

Scaffolder

Scaffolder is written in PHP scripting language by using the object-oriented programming (OOP) paradigm. The version used is PHP 5.2. Data has been stored in a relational database. PostgreSQL is the database management system (DBMS) selected and installed to manage all the Scaffolder data. Indexing is heavily used for providing quick access to data.

Finding links between contigs

- 1) Identification of links by contig-end similarity was performed by developing and running an ad-hoc PHP script, which is able to detect, within a pool of given sequences, those ending with the same stretch of N bps, with N varying from 10 to 50.
- 2) Connections by coding information were found by running BLASTX (e-value cut-off 0.01) on all 100 bps contig-ends against all known bacterial proteins annotated in KEGG (release 39.0). Two contig-ends were considered as connected when they match by BLASTX different regions of at least one common protein at a maximum distance of 30 aminoacids.
- 3) Links by analysis of initial reads were searched by running BLASTN (e-value cut-off 0.01, minimum match ≥ 30 bps) on all 100 bps contig-ends against all sequenced primary reads and considering as connected those contig-ends sharing at least one same matching read.

Building the connected graph

Connected graphs are done by using *dot*, a tool of the graphViz package. A specific module was developed in order to convert scaffold data and links in a suitable format for dot. The same module generates both graph images and html clickable maps. Scaffolds are represented in maps as colored boxes whose width is proportional to contig length. Contigs are colored according to the following criteria:

When one contig has the same number of reads at the ends:

- Gray: contig without connections
- Light green: contig with expected coverage and with one link per end
- Dark green: contig with expected coverage but with only one end linked
- Yellow: contig with multiple coverage and with the same number of multiple links per ends according to the coverage.
- Orange: contig with multiple coverage, but with a higher number of links than expected
- Light red: contig with a lower coverage than expected and with the same number of multiple links per ends
- Dark red: contig with multiple coverage, but with a lower number of links than expected

Support in design of PCR experiments

PCR primer design was set up by merging the ends of connected contigs, also taking into account their putative orientation. Primers were designed by using the eprimer3 tool from the EMBOSS package on each end by using all other contig ends as mispriming library (options `-mispriminglibraryfile`) in order to avoid recognition of other ends. Further options, such as minimum GC content and minimum length, were used as default. PrimerSearch program, also available in EMBOSS, was used to simulate PCR experiments with the identified primers, using a tolerance of 20% of mismatch between primers and target sequence.

Primers identified for the X model were selected for experiments when a unique amplificate per combination was predicted.

Primers identified for the Y model were selected for experiments when only two amplificates per combination were predicted.

Amplificate sequences were aligned to each combination of scaffold-ends by using bl2seq: in this way alignments were evaluated to confirm the PCR results and correct the sequence of repeated scaffolds, if necessary.

Aligning initial reads to a reference sequence

Reads matching a reference sequence such as a contig or a scaffold sequence, were selected from the initial pool by using BLAST (setting e-value cut off 0.01, allowing the search without filtering) and discarding all matches shorter than 20bps. Reads matching artificial sequences derived from combination of ends were also found in this way. Alignment of selected reads to the reference sequence was done by using Blastalign, modified in order to launch MEGABLAST instead of BLASTN and setting the maximum proportion of allowed gaps in every sequence to 0.99. Aligned sequences were sorted in order to display matching reads with 5' to 3' order.

Display alignment

A PHP script was developed to draw pictures of alignments where presence of heterogeneities are highlighting. The script looks through each column of the alignment and when it finds more than one base, differently colors base variants.

Micro-heterogeneities analysis

Reads matching for a scaffold (or contig) sequence are detected and aligned as described in the previous paragraph. A PHP script was developed to analyze the alignment and find hidden sequence variants within the reference sequences.

References

Papers

- Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: Basic local alignment search tool. *J Mol Biol* 1990, 215:403-410.
- Bachellier S, Clément JM, Hofnung M. Short palindromic repetitive DNA elements in enterobacteria: a survey. *Res Microbiol.* 1999 Nov-Dec;150(9-10):627-39. Review.
- Barrangou R, Fremaux C, Deveau H, Richards M, Boyaval P, Moineau S, Romero DA, Horvath P. CRISPR provides acquired resistance against viruses in prokaryotes. *Science.* 2007 Mar 23;315(5819):1709-12.
- Bateman A, Birney E, Durbin R, Eddy SR, Finn RD, Sonnhammer EL. Pfam 3.1: 1313 multiple alignments and profile HMMs match the majority of proteins. *Nucleic Acids Res.* 1999 Jan 1;27(1):260-2.
- Belshaw R, Katzourakis A. BlastAlign: a program that uses blast to align problematic nucleotide sequences. *Bioinformatics.* 2005 Jan 1;21(1):122-3.
- Berezikov E, Guryev V, van de Belt J, Wienholds E, Plasterk RH, Cuppen E. Phylogenetic shadowing and computational identification of human microRNA genes. *Cell.* 2005 Jan 14;120(1):21-4.
- Berg BL, Baron C, Stewart V. Nitrate-inducible formate dehydrogenase in *Escherichia coli* K-12. II. Evidence that a mRNA stem-loop structure is essential for decoding opal (UGA) as selenocysteine. *J Biol Chem.* 1991 Nov 25;266(33):22386-91.
- Bonnet E, Wuyts J, Rouzé P, Van de Peer Y. Evidence that microRNA precursors, unlike other non-coding RNAs, have lower folding free energies than random sequences. *Bioinformatics.* 2004 Nov 22;20(17):2911-7.
- Burge C, Karlin S. Prediction of complete gene structures in human genomic DNA. *J Mol Biol.* 1997 Apr 25;268(1):78-94.
- Chaisson MJ, Pevzner PA. Short read fragment assembly of bacterial genomes. *Genome Res.* 2008 Feb;18(2):324-30.
- Claverie JM, Ogata H. The insertion of palindromic repeats in the evolution of proteins. *Trends Biochem Sci.* 2003 Feb;28(2):75-80.

- Coburn GA, Mackie GA. Degradation of mRNA in *Escherichia coli*: an old problem with some new twists. *Prog Nucleic Acid Res Mol Biol*. 1999;62:55-108. Review.
- Davidsen T, Rødland EA, Lagesen K, Seeberg E, Rognes T, Tønjum T. Biased distribution of DNA uptake sequences towards genome maintenance genes. *Nucleic Acids Res*. 2004 Feb 11;32(3):1050-8.
- De Gregorio E, Abrescia C, Carlomagno MS, Di Nocera PP: Ribonuclease III-mediated processing of specific *Neisseria meningitidis* mRNAs. *Biochem J* 2003, 374:799-805.
- De Gregorio E, Silvestro G, Petrillo M, Carlomagno MS, Di Nocera PP. Enterobacterial repetitive intergenic consensus sequence repeats in *Yersinia*: genomic organization and functional properties. *J Bacteriol*. 2005 Dec;187(23):7945-54.
- De Gregorio E, Silvestro G, Petrillo M, Carlomagno MS, Di Nocera PP. Enterobacterial repetitive intergenic consensus sequence repeats in *Yersinia*: genomic organization and functional properties. *J Bacteriol*. 2005 Dec;187(23):7945-54.
- Eddy SR, Durbin R. RNA sequence analysis using covariance models. *Nucleic Acids Res*. 1994 Jun 11;22(11):2079-88.
- Engelhorn M, Boccard F, Murtin C, Prentki P, Geiselman J. In vivo interaction of the *Escherichia coli* integration host factor with its specific binding sites. *Nucleic Acids Res*. 1995 Sep 11;23(17):2959-65.
- Enright AJ, Van Dongen S, Ouzounis CA. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res*. 2002 Apr 1;30(7):1575-84.
- Ermolaeva MD, Khalak HG, White O, Smith HO, Salzberg SL. Prediction of transcription terminators in bacterial genomes. *J Mol Biol*. 2000 Aug 4;301(1):27-33.
- Espéli O, Boccard F. In vivo cleavage of *Escherichia coli* BIME-2 repeats by DNA gyrase: genetic characterization of the target and identification of the cut site. *Mol Microbiol*. 1997 Nov;26(4):767-77.
- Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF, Dougherty BA, Merrick JM, et al.
- Science. 1995 Jul 28;269(5223):496-512. Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science*. 1995 Jul 28;269(5223):496-512.

- Freyhult E, Gardner PP, Moulton V. A comparison of RNA folding measures. *BMC Bioinformatics*. 2005 Oct 3;6:241.
- Gardner PP, Giegerich R. A comprehensive comparison of comparative RNA structure prediction approaches. *BMC Bioinformatics*. 2004 Sep 30;5:140.
- Gilson E, Bachellier S, Perrin S, Perrin D, Grimont PA, Grimont F, Hofnung M. Palindromic unit highly repetitive DNA sequences exhibit species specificity within Enterobacteriaceae. *Res Microbiol*. 1990 Nov-Dec;141(9):1103-16.
- Gilson E, Saurin W, Perrin D, Bachellier S, Hofnung M. The BIME family of bacterial highly repetitive sequences. *Res Microbiol*. 1991 Feb-Apr;142(2-3):217-22.
- Gorodkin J, Heyer LJ, Stormo GD. Finding the most significant common sequence and structure motifs in a set of RNA sequences. *Nucleic Acids Res*. 1997 Sep 15;25(18):3724-32.
- Henkin TM, Yanofsky C. Regulation by transcription attenuation in bacteria: how RNA provides instructions for transcription termination/antitermination decisions. *Bioessays*. 2002 Aug;24(8):700-7. Review.
- Higgins CF, McLaren RS, Newbury SF. Repetitive extragenic palindromic sequences, mRNA stability and gene expression: evolution by gene conversion? A review. *Gene*. 1988 Dec 10;72(1-2):3-14. Review.
- Himmelreich R, Hilbert H, Plagens H, Pirkel E, Li BC, Herrmann R. Complete sequence analysis of the genome of the bacterium *Mycoplasma pneumoniae*. *Nucleic Acids Res*. 1996 Nov 15;24(22):4420-49.
- Höchsmann M, Töller T, Giegerich R, Kurtz S. Local similarity in RNA secondary structures. *Proc IEEE Comput Soc Bioinform Conf*. 2003;2:159-68.
- Hofacker IL, Fekete M, Stadler PF. Secondary structure prediction for aligned RNA sequences. *J Mol Biol*. 2002 Jun 21;319(5):1059-66.
- Huang X, Wang J, Aluru S, Yang SP, Hillier L. PCAP: a whole-genome assembly program. *Genome Res*. 2003 Sep;13(9):2164-70.
- Johansson J, Mandin P, Renzoni A, Chiaruttini C, Springer M, Cossart P. An RNA thermosensor controls expression of virulence genes in *Listeria monocytogenes*. *Cell*. 2002 Sep 6;110(5):551-61.
- Kampa D, Cheng J, Kapranov P, Yamanaka M, Brubaker S, Cawley S, Drenkow J, Piccolboni A, Bekiranov S, Helt G, Tammana H, Gingeras TR. Novel RNAs

- identified from an in-depth analysis of the transcriptome of human chromosomes 21 and 22. *Genome Res.* 2004 Mar;14(3):331-42.
- Kanehisa M, Goto S. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.* 2000 Jan 1;28(1):27-30.
 - Kazantsev AV, Pace NR. Bacterial RNase P: a new view of an ancient enzyme. *Nat Rev Microbiol.* 2006 Oct;4(10):729-40. Review.
 - Kingsford CL, Ayanbule K, Salzberg SL. Rapid, accurate, computational discovery of Rho-independent transcription terminators illuminates their relationship to DNA uptake. *Genome Biol.* 2007;8(2):R22.
 - Klein RJ, Eddy SR. RSEARCH: finding homologs of single structured RNA sequences. *BMC Bioinformatics.* 2003 Sep 22;4:44.
 - Laslett D, Canback B, Andersson S. BRUCE: a program for the detection of transfer-messenger RNA genes in nucleotide sequences. *Nucleic Acids Res.* 2002 Aug 1;30(15):3449-53.
 - Leplae R, Hebrant A, Wodak SJ, Toussaint A: ACLAME: a CLAssification of Mobile genetic Elements. *Nucleic Acids Res* 2004,32:D45-49.
 - Lim LP, Glasner ME, Yekta S, Burge CB, Bartel DP. Vertebrate microRNA genes. *Science.* 2003 Mar 7;299(5612):1540.
 - Lowe TM, Eddy SR. A computational screen for methylation guide snoRNAs in yeast. *Science.* 1999 Feb 19;283(5405):1168-71.
 - Lowe TM, Eddy SR. tRNAscan-SE: a program for improved detection of transfer RNA genes in genomic sequence. *Nucleic Acids Res.* 1997 Mar 1;25(5):955-64.
 - Macke TJ, Ecker DJ, Gutell RR, Gautheret D, Case DA, Sampath R. RNAMotif, an RNA secondary structure definition and search algorithm. *Nucleic Acids Res.* 2001 Nov 15;29(22):4724-35.
 - Mahillon J, Chandler M. Insertion sequences. *Microbiol Mol Biol Rev.* 1998 Sep;62(3):725-74. Review.
 - Majoros WH, Pertea M, Salzberg SL. TigrScan and GlimmerHMM: two open source ab initio eukaryotic gene-finders. *Bioinformatics.* 2004 Nov 1;20(16):2878-9. Epub 2004 May 14.
 - Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, Bemben LA, Berka J, Braverman MS, Chen YJ, Chen Z, Dewell SB, Du L, Fierro JM, Gomes XV, Godwin BC, He W, Helgesen S, Ho CH, Irzyk GP, Jando SC, Alenquer ML, Jarvie TP, Jirage KB, Kim JB, Knight JR, Lanza JR, Leamon JH, Lefkowitz SM, Lei M,

- Li J, Lohman KL, Lu H, Makhijani VB, McDade KE, McKenna MP, Myers EW, Nickerson E, Nobile JR, Plant R, Puc BP, Ronan MT, Roth GT, Sarkis GJ, Simons JF, Simpson JW, Srinivasan M, Tartaro KR, Tomasz A, Vogt KA, Volkmer GA, Wang SH, Wang Y, Weiner MP, Yu P, Begley RF, Rothberg JM. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*. 2005 Sep 15;437(7057):376-80.
- Martínez-Abarca F, Toro N. Group II introns in the bacterial world. *Mol Microbiol*. 2000 Dec;38(5):917-26. Review.
 - Mathews DH, Disney MD, Childs JL, Schroeder SJ, Zuker M, Turner DH. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of RNA secondary structure. *Proc Natl Acad Sci U S A*. 2004 May 11;101(19):7287-92.
 - Mathews DH, Turner DH. Dynalign: an algorithm for finding the secondary structure common to two RNA sequences. *J Mol Biol*. 2002 Mar 22;317(2):191-203.
 - Mazzone M, De Gregorio E, Lavitola A, Pagliarulo C, Alifano P, Di Nocera PP. Whole-genome organization and functional properties of miniature DNA insertion sequences conserved in pathogenic *Neisseriae*. *Gene*. 2001 Oct 31;278(1-2):211-22.
 - McMurray CT. DNA secondary structure: a common and causative factor for expansion in human disease. *Proc Natl Acad Sci U S A*. 1999 Mar 2;96(5):1823-5. Review.
 - Merino E, Yanofsky C. Transcription attenuation: a highly conserved regulatory strategy used by bacteria. *Trends Genet*. 2005 May;21(5):260-4.
 - Nudler E, Mironov AS. The riboswitch control of bacterial metabolism. *Trends Biochem Sci*. 2004 Jan;29(1):11-7. Review.
 - Nudler E, Mironov AS. The riboswitch control of bacterial metabolism. *Trends Biochem Sci*. 2004 Jan;29(1):11-7. Review.
 - Nussinov R, Jacobson AB. Fast algorithm for predicting the secondary structure of single-stranded RNA. *Proc Natl Acad Sci U S A*. 1980 Nov;77(11):6309-13.
 - Petrillo M, Silvestro G, Di Nocera PP, Boccia A, Paoletta G: Stemloop structures in prokaryotic genomes. *BMC Genomics* 2006,7:170.
 - Pop M, Kosack DS, Salzberg SL. Hierarchical scaffolding with Bambus. *Genome Res*. 2004 Jan;14(1):149-59.

- Richter DC, Ott F, Auch AF, Schmid R, Huson DH. MetaSim: a sequencing simulator for genomics and metagenomics. PLoS ONE. 2008 Oct 8;3(10):e3373.
- Rivas E, Eddy SR. Noncoding RNA gene detection using comparative sequence analysis. BMC Bioinformatics. 2001;2:8.
- Rouquette-Loughlin CE, Balthazar JT, Hill SA, Shafer WM: Modulation of the mtrCDE-encoded efflux pump gene complex of *Neisseria meningitidis* due to a Correia element insertion sequence. Mol Microbiol 2004, 54:731-741.
- Rowe-Magnus DA, Guerout AM, Biskri L, Bouige P, Mazel D. Comparative analysis of superintegrons: engineering extensive genetic diversity in the Vibrionaceae. Genome Res. 2003 Mar;13(3):428-42.
- Salzberg SL, Yorke JA. Beware of mis-assembled genomes. Bioinformatics. 2005 Dec 15;21(24):4320-1.
- Sankoff, D. (1985) Simultaneous solution of the RNA folding, alignment and protosequence problems. SIAM J. Appl. Math., 45, 810–825.
- Schattner P, Decatur WA, Davis CA, Ares M Jr, Fournier MJ, Lowe TM. Genome-wide searching for pseudouridylation guide snoRNAs: analysis of the *Saccharomyces cerevisiae* genome. Nucleic Acids Res. 2004 Aug 11;32(14):4281-96.
- Sundquist A, Ronaghi M, Tang H, Pevzner P, Batzoglou S. Whole-genome sequencing and assembly with high-throughput, short-read technologies. PLoS ONE. 2007 May 30;2(5):e484.
- Tinoco I Jr, Bustamante C. How RNA folds. J Mol Biol. 1999 Oct 22;293(2):271-81. Review.
- van Belkum A, Scherer S, van Alphen L, Verbrugh H. Short-sequence DNA repeats in prokaryotic genomes. Microbiol Mol Biol Rev. 1998 Jun;62(2):275-93. Review.
- Washietl S, Hofacker IL, Stadler PF. Fast and reliable prediction of noncoding RNAs. Proc Natl Acad Sci U S A. 2005 Feb 15;102(7):2454-9.
- Washietl S, Pedersen JS, Korbelt JO, Stocsits C, Gruber AR, Hackermüller J, Hertel J, Lindemeyer M, Reiche K, Tanzer A, Ucla C, Wyss C, Antonarakis SE, Denoeud F, Lagarde J, Drenkow J, Kapranov P, Gingeras TR, Guigó R, Snyder M, Gerstein MB, Reymond A, Hofacker IL, Stadler PF. Structured RNAs in the ENCODE selected regions of the human genome. Genome Res. 2007 Jun;17(6):852-6

- Washietl S. Prediction of structural noncoding RNAs with RNAz. *Methods Mol Biol.* 2007;395:503-26.
- Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science.* 1995 Jul 28;269(5223):496-512.
- Workman C, Krogh A. No evidence that mRNAs have lower folding free energies than random sequences with the same dinucleotide distribution. *Nucleic Acids Res.* 1999 Dec 15;27(24):4816-22.
- Xu Y, Mural R, Shah M, Uberbacher E. Recognizing exons in genomic sequence using GRAIL II. *Genet Eng (N Y).* 1994;16:241-53.
- Zuker M, Stiegler P. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.* 1981 Jan 10;9(1):133-48.
- Zuker M. On finding all suboptimal foldings of an RNA molecule. *Science.* 1989 Apr 7;244(4900):48-52. Review.

Websites

454 Life Science - www.454.com

Assembly archive - <http://www.ncbi.nlm.nih.gov/Traces/assembly/assmbrowser.cgi>

Blastalign - <http://www.bio.ic.ac.uk/research/belshaw/BlastAlign.tar>

Complete genomics www.completegenomics.com

Emboss - <http://emboss.sourceforge.net/>

Encode project - www.genome.gov/10005107

Ensembl - www.ensembl.org

European Read Archive at - <http://www.ebi.ac.uk/embl/Documentation/ENA-Reads.html>

GenBank - www.ncbi.nlm.nih.gov/Genbank

Graphviz tool - www.graphviz.org

Helicosbio - www.helicosbio.com

Illumina - www.illumina.com

KEGG Automatic Annotation Server KAAS - <http://www.genome.jp/kegg/kaas/>

PHP – <http://php.net>

Phrap - www.phrap.org

Postgresql - www.postgresql.org/

Read Archive - http://www.ddbj.nig.ac.jp/sub/trace_sra-e.html

Short Read Archive - <http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi>

Swiss-Prot - www.expasy.org/sprot

TIGR - www.tigr.org

Acknowledgments

I wish to express my gratitude and appreciation to the many persons whose ideas and help have strongly contributed to the development of this work. I wish to thank professor Francesco Salvatore, who has carefully followed my progression during these years. I am really grateful to professor Giovanni Paoletta, who has followed the development of my studies with continuous attention, supporting my growth with precious suggestions and contributions.

In particular, of all the colleagues, I wish to thank Mauro Petrillo, who has always given important contributions in developing and implementation of Scaffold and the pipeline to analyze bacterial SLs. He is a good colleague and friend and his help was fundamental during my PhD period. I wish to thank professor Pier Paolo Di Nocera who supports me in analyses of stem-loop families in bacterial genomes and professor Toby Gibson for kindly act as external tutor.

I wish to thank Gianluca Busiello and Angelo Boccia for their help and precious suggestions. Leandra Sepe and Concita Cantarella have been precious colleagues and important friends.

Finally I wish to thank my family and my girlfriend, who have supported me since the beginning.